Extraction of Buildings from High-Resolution Satellite Data and Airborne LIDAR

Gunho Sohn
Department of Geomatic Engineering
University College London

Supervisor:
Prof. Ian J. Dowman

Thesis submitted for the degree of Doctor of Philosophy (Ph.D.)
at the University of London

April 2004

ABSTRACT

Automatic building extraction is a difficult object recognition problem due to a high complexity of the scene content and the object representation. There is a dilemma to select appropriate building models to be reconstructed; the models have to be generic in order to represent a variety of building shape, whereas they also have to be specific to differentiate buildings from other objects in the scene. Therefore, a scientific challenge of building extraction lies in constructing a framework for modelling building objects with appropriate balance between generic and specific models. This thesis investigates a synergy of IKONOS satellite imagery and airborne LIDAR data, which have recently emerged as powerful remote sensing tools, and aims to develop an automatic system, which delineates building outlines with more complex shape, but by less use of geometric constraints.

The method described in this thesis is a two step procedure: building detection and building description. A method of automatic building detection that can separate individual buildings from surrounding features is presented. The process is realized in a hierarchical strategy, where terrain, trees, and building objects are sequentially detected. Major research efforts are made on the development of a LIDAR filtering technique, which automatically detects terrain surfaces from a cloud of 3D laser points. The thesis also proposes a method of building description to automatically reconstruct building boundaries. A building object is generally represented as a mosaic of convex polygons. The first stage is to generate polygonal cues by a recursive intersection of both data-driven and model-driven linear features extracted from IKONOS imagery and LIDAR data. The second stage is to collect relevant polygons comprising the building object and to merge them for reconstructing the building outlines. The developed LIDAR filter was tested in a range of different landforms, and showed good results to meet most of the requirements of DTM generation and building detection. Also, the implemented building extraction system was able to successfully reconstruct the building outlines, and the accuracy of the building extraction is good enough for mapping purposes.

To Meeyoung and Seiyon

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENT

I would like to thank my supervisor Prof. Ian Dowman for his boundless support, encouragement and guidance throughout this study. I has been always challenged and motivated by his invaluable advice and suggestions.

This thesis would not have been possible without the contribution made by many researchers: Prof. Peter Muller and Dr. Muki Haklay at the Department of Geomatic Engineering of the UCL, who kindly provided IKONOS images, LIDAR data, and OS MasterMap® for this research; Dr. Yiorgos Chrysanthou and In-Su Yu at the Department of Computer Science of the UCL engaged me in useful discussions relating to this work; Dr. Guang-Zhong Yang at the Department of Computing of Imperial College provided programming resources for implementing an image-based local Fourier analysis.

I must express my gratitude to all the staffs and friends at the UCL Department of Geomatic Engineering for sharing their experience and insights into this subject.

I also would like to thank to QinetiQ, the UCL Graduate School and the Centre for Scientific Enterprise Ltd., who have financially supported to complete this work.

Finally I would specially like to thank to my parents and my brother, without whose support, encouragement, and love, this work would have been impossible.

CHAPTER 1
INTRODUCTION

1.1 Problem domain

One of the ultimate aims in photogrammetry is to generate an urban landscape model (ULM) to show the objects and landcover of an urban area in three dimensions (Dowman, 2000). This task requires *object extraction*, which detects the object of interest and describes its geometric boundary, usually in 3D, from remotely sensed data. With accurate boundary representation of an object, more complex tasks such as *object reconstruction* (modelling 3D structures of the rooftop) and *object recognition* (assigning semantic description to the object extracted) can be achieved.

A very large number of natural and cultural objects could be involved in constructing the ULM. Amongst them, the building object has been regarded as the most prominent feature in an urban environment. It is widely agreed that acquiring accurate and up-to-date representation of buildings is fundamental for a variety of applications. These include urban planning, environment studies, risk assessment, emergency services, facility management, wireless telecommunication planning, transportation, tourism, video games, change detection, and mission planning for military purpose (Ameri, 2000; Danahy, 1999; Lange, 1999; Leberl et al., 1999).

Traditionally, 3D building extraction has been achieved by mainly relying on manual photo-interpretation with the support of DPWs (Digital Photogrammetric Workstations). Although it has been widely used in practice, the manual photogrammetric method has largely proven to be inefficient, especially in the case that a huge amount of data must be processed. Manual 3D reconstruction of buildings still remains in a tedious and

expensive process, and requires highly qualified personnel with specific knowledge of a site landscape, sensors used, and complex data processing. Thus, the full automation of 3D reconstruction of buildings has been regarded as a grand scientific challenge. Over the last decade, the problems associated with 3D building extraction from remotely sensed data have been central in many journals and research workshops (Baltsavias et al., 1995; Baltsavias et al., 2001; CVIU, 1998; Forstner & Plumer, 1997; Gruen et al., 1997).

Fundamentally, 3D building extraction can be achieved through certain hierarchical processing stages. The first stage starts with extracting cues, which are locally distinguishable geometric, (e.g., edges or corners) or chromatic, properties extracted by low-level feature extraction algorithms from "view-centered" data (*cue extraction*). Then, those cues that are related only to a building are found (*building detection*), and the final stage ends up with reconstructing the building boundaries in an object-centered 3D model (*building description*). The development of an adequate building extraction system depends on the success of all the tasks mentioned above. Although a lot of algorithms and systems have been proposed towards solving this problem, success of fully automated reconstruction of building objects from remotely sensed data is still far to reach, and only partial solutions and limited success in constrained environments have been reported. There are several reasons to explain this difficulty.

Firstly, urban scenes usually provide very rich scene content, and they contain a certain amount of information that is not relevant to the given task of building extraction. The scene complexity posed by an urban environment makes automatic building detection difficult. The term *building detection* is used for the procedure of discriminating buildings from their surroundings and of obtaining the building as a single object. Several cues, edges or corners for example, are used to solve this problem, and a building object is localized as its relevant cues are found. In an urban scene, however, a huge amount of edges could be extracted from vegetation, vehicles, roads, and street furniture as well as buildings. It is hard to distinguish whether an edge belongs to a building, its shadow, or other objects. Some heuristic knowledge can be used to differentiate building cues from other distracting features. For instance, it is assumed that building cues can be found in an area with certain brightness or colours or nearby shadow. Yet, those heuristics can be applied only in constrained environment, and there

are many situations in complex urban areas where the pre-specified heuristics are not valid.

Secondly, *building description* is a geometric representation of building boundary, which can be reconstructed with geometric cues extracted from the data. The problem of building description is caused by the fact that there is a significant loss of relevant building cues due to occlusion, poor contrast, shadows and disadvantageous image perspective. Under this circumstance, the geometric cues may be highly fragmented or completely missed, and the building object can be only partially described with such incomplete cues. Hence, the building description requires a methodology to recover those fragmented or missing cues in order to fully describe the building boundary. However, recovering a full description of building shape from incomplete cues is an "ill-posed" problem, which needs to be converted into a "well-posed" problem by employing supporting building models for generally representing the shape of the building object.

In general, there are two building models for the building description: *specific model* and *generic model*. The specific building modelling assumes that most of building outlines can be reconstructed by simple rectangle, or parallelogram or its combinations. As a corresponding building model is fitted to the observed data and is verified with certain evidences, the building boundary can be reconstructed by the selected models. On the other hand, the generic building modelling reconstructs the building shape without using pre-specified models, but by densely collecting geometric cues comprising all parts of building boundaries. This approach relies mainly on low-level cue extraction algorithms and uses multiple data frames, capturing the same scene with different viewing angles and illuminating conditions. It is assumed that fragmented or missing cues in the one data frame can be recovered from the others with better contrast and perspective views avoiding shadows and occlusions.

Both approaches mentioned above have their own pros and cons in the building description. The benefit of the former method is to reconstruct the building boundary with minimal use of cues, which is a useful property, especially in the case that sufficient cue extraction is difficult to obtain with given data sources. However, the method is not suitable for representing a variety of building shapes. In fact, there are

enormous variations in the shape of the buildings in urban scenes. The latter method is useful to delineate more complex building shape with no limitation, if sufficient cues can be collected from multiple data frames. However, it is difficult to answer the question how many data frames are required in order to obtain such high-quality cues, and sometimes the acquisition of even two data frames is not always available. Therefore, a scientific challenge for the building description lies in developing techniques to delineate complex shapes of the building object with minimal use of constrained models and data sources.

Thirdly, the building extraction system is highly sensor dependent, and its development is influenced by the data sources used. The reason for this is that the primary data used for a building extraction system determines which knowledge can be available as primary cues for representing generic shape of the building object. Algorithms for the detection and description of buildings are differently implemented, depending on which primary cues are used. For instance, a building extraction system using 3D depth information is much different from a system that relies on only monocular imagery. The primary data to support the building extraction is available from a variety of sensors, which includes spaceborne or airborne optical sensors (with panchromatic or multi-spectral bands), laser scanners (with pulse or continuous wave), microwave imaging sensors (SAR or Interferometric SAR), or existing 2D ground plans. Each data source has its own strength and weakness as the primary cue for the building extraction. An important research issue is to fuse different data sources so that the weaknesses of individual cues are compensated for by others. However, it is a complicated task to determine which cues must be selected in the fusion process and how to combine them at each process of *cue extraction*, *building detection* and *building description*.

For a long time, high-resolution aerial images, usually less than 50 cm, have been mainly targeted as the primary data for exploiting the building extraction. Recent development of remote sensing technologies diversifies the methods for acquiring the primary data in building extraction. Such new comers are IKONOS satellite and LIDAR (Light Detection and Ranging). Over the last five years, both sensors have become powerful tools to replace or complement conventional data acquisition techniques. The rapid emergence of IKONOS satellite and LIDAR has created an urgent need for new techniques for 3D building extraction.

IKONOS, the world's first commercial high-resolution imaging satellite, was successfully launched in September of 1999, and has been consistently providing high quality 1-meter panchromatic and 4-meter multi-spectral images. Unlike aerial data, IKONOS imagery can be acquired for any location with minimal bureaucratic restriction, and offers unprecedented access to accurate and timely information. Compared to classic LANDSAT and SPOT, the detail distinguishable from finer resolution of IKONOS satellite images provide enormous benefit in mapping and modelling urban area. In particular, in countries where there is little experience in mapping and aerial photography, and an absence of the infrastructure required to collect and process such data, IKONOS imagery can provide a rapid, high-quality data source for the production of image maps and vector maps of various levels of complexity (Holland & Marshall, 2003). For these reasons, the building extraction from IKONOS imagery has recently attracted increasing research interests.

Since the mid 1990s, LIDAR was developed as an alternative technique of traditional aerial photogrammetry and used for direct measurement of the topographical terrain surface. Mounted in an airborne platform, LIDAR uses laser light to accurately measure heights of points on the ground or non-ground objects and rapidly generates densely spaced elevation data. The most attractive feature of LIDAR is its superior vertical accuracy and high point density. Such high-quality 3D information is still difficult to automatically obtain by the aerial photogrammetry. This characteristic of the LIDAR data is particularly well suited for detection and modelling of 3D building objects.

Much research efforts have been made to automatically detect and describe buildings from either IKONOS or LIDAR, but further investigation is needed to develop a fully operational building extraction system. This is caused by the weakness inherited from each sensor. The spatial resolution of IKONOS imagery is still many times poorer than aerial images. This characteristic of IKONOS imagery makes traditional cue extraction algorithms difficult to apply appropriately. A consequence is a significant loss of geometric cues which leads to adverse effects in building detection and building description. On the other hand, LIDAR data has a critical drawback in building description. Since LIDAR measures heights of irregularly spaced points, it is a major bottleneck to extract break lines for delineating building boundaries. For this reason,

LIDAR has been mainly used for building reconstruction, where the problem of building description is avoided by the use of existing 2D ground plans.

However, an efficient procedure of data fusion can compensate for the weaknesses of both IKONOS imagery and LIDAR data for building extraction. It is because they naturally have the complementary properties of each other; IKONOS imagery can provide rich scene content with a variety of information such as intensity, colours, lines and corners, while LIDAR can attribute accurate 3D information to those cues extracted from IKONOS imagery. Many researchers highlighted the synergy of IKONOS imagery and LIDAR data for building extraction and reconstruction. To realize this, state-of-the-art in remote sensing technology combines optical sensors with LIDAR, which is commercially available in today's market. However, studies of the data fusion to combine both sensors for building extraction are so far less reported. Therefore, the development of building extraction system using both sensors needs further refinement in methodology to tackle the problems discussed so far.

1.2 Research objectives

The main aims of this research are to understand the technologies of IKONOS satellite and airborne LIDAR system, which are recently introduced in photogrammetry, and to develop a building extraction system that automatically detects the building objects and represents their boundary information in an urban environment using IKONOS satellite imagery and airborne LIDAR data. The research is mainly concerned about investigating the inherited limitation of each sensor and introducing the methodologies to maximize the synergy achieved by fusing IKONOS satellite imagery and airborne LIDAR data for extracting the buildings. In particular, two research interests are exploited to develop the automatic building extraction system.

Firstly, the research aims to develop a method to effectively reduce the scene complexity, which is common in urban areas, and thus, the building extraction process can be applied to focused areas where only individual buildings are located. This task requires removing distracting features which are not related to the building structures. Thus, the research aims to develop methods of automatically detecting and classifying the terrain surface, low objects, vegetated objects, and building objects.

Secondly, the research introduces the new concepts and develops robust methods to combine a *data-driven* and a *model-driven* approach for the representation of building boundaries. The term *data-driven* is used to indicate that the process of description of building shapes is performed without *a prior* knowledge of the building type or its structure, while the *model-driven* uses pre-specified building models. The proposed approach aims to manage buildings with different shapes and complexities (except buildings with curvatures). In particular, the method focuses on a problematic situation, where sufficient cues for describing a complex building shape are difficult to obtain from monocular IKONOS imagery. Therefore, the research aims to develop new methods of cue generation and cue grouping through several fusion techniques, where the data-driven and model-driven approaches are combined and primary cues extracted from IKONOS imagery and LIDAR data are merged.

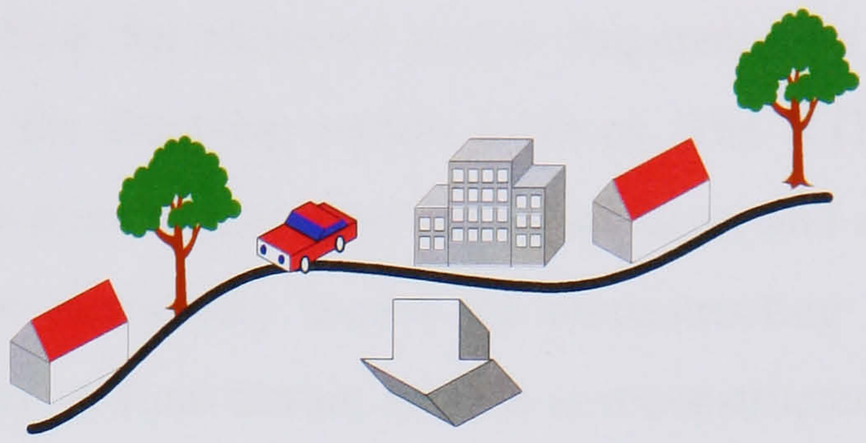
1.3 Method overview

Figure 1-1 illustrates a workflow of the major components and their subsequent processes in the automated building extraction system proposed in this thesis. The system requires IKONOS imagery with multi-spectral bands, where each image pixel is resampled at 1-metre on the ground and LIDAR data, which planimetric point spacing is approximately 3-metres in average. It is pre-assumed that both data are geo-referenced. The building extraction system is formulated into two major processes; the "building detection process" and the "building boundary reconstruction process".

The building detection process aims to sequentially detect and remove dominant features in ULM including the terrain surface, low objects, and trees, and thus to finally localize the buildings in the IKONOS imagery and LIDAR space. This process allows the subsequent building boundary reconstruction process to focus only on one building structure, which can greatly reduce the scene complexity, and simplify the reconstruction process.

Data sources

- IKONOS geo-referenced imagery with multi-spectral bands
- LIDAR X,Y,Z points



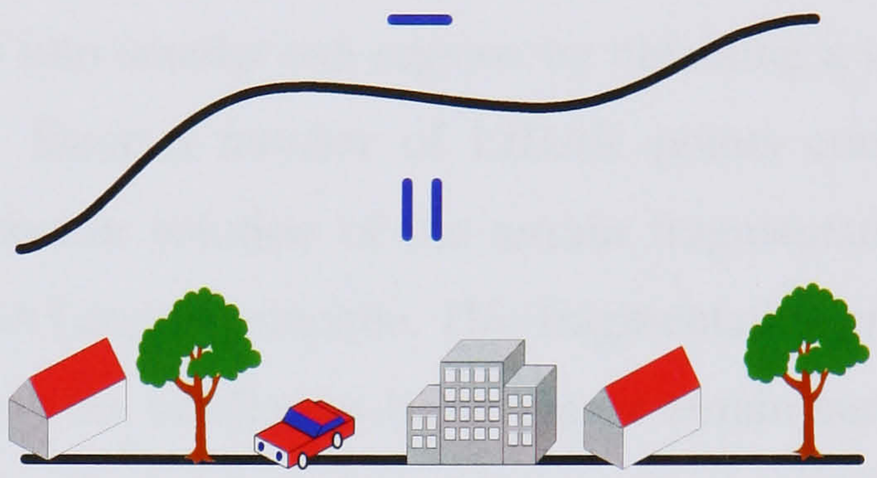
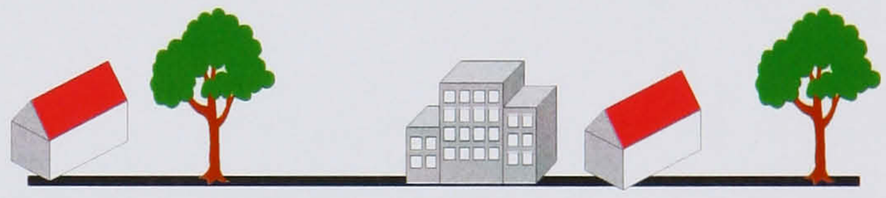
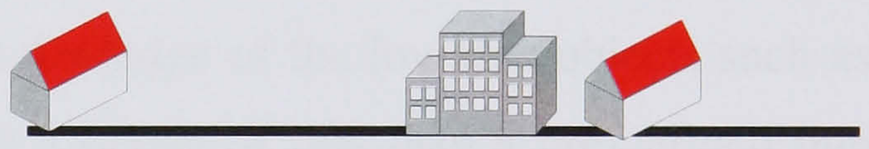
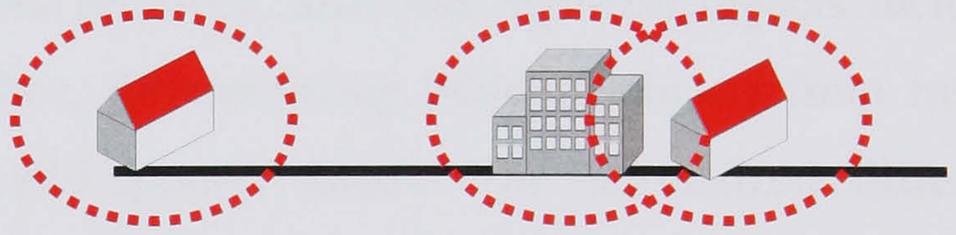
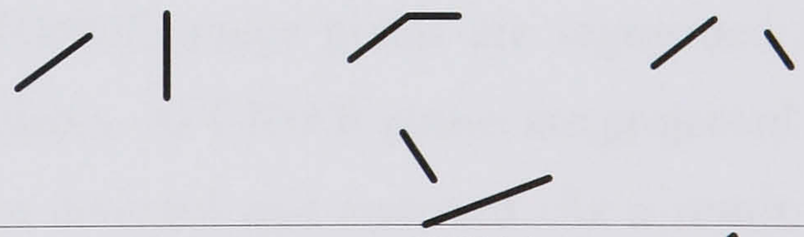
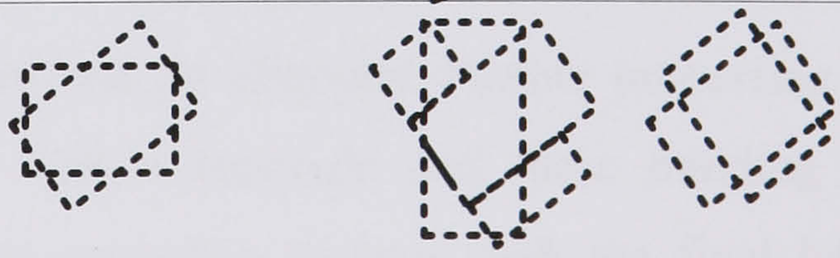
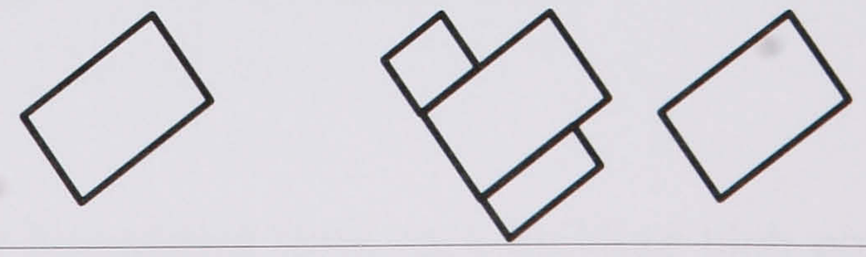
Major Processes	Sub-processes	Output
Building detectior	Terrain detection and removal	
	Low-rise object detection and removal	
	Tree detection and removal	
	Building blob detection	
Building boundary description	Data-driven feature extraction	
	Model-driven feature extraction	
	Binary Space Partitioning (BSP)	

Figure 1-1. Schematic workflow of the overall process of the proposed building extraction system

In the first stage of the building detection, the recursive terrain fragmentation (RTF) filter is applied to the LIDAR points for detecting terrain surfaces. The RTF filter assumes that a generic terrain surface is a set of piecewise planar surfaces, and a plane terrain model (PTM) is defined as an elementary model for reconstructing terrain surfaces from the LIDAR measurements. A final terrain surface is reconstructed in an iterative hypothesis-test scheme. The entire LIDAR space is initially hypothesized by a single PTM. If the hypothesized area does not satisfy the conditions of the PTM, the initial terrain model will be fragmented into smaller sub-regions by obtaining a terrain point from underlying LIDAR points. Since a number of LIDAR points could be considered as the terrain point, an optimum solution of the terrain fragmentation is achieved by MDL (Minimum Description Length) principle. This fragmentation process continues until all the fragmented regions are verified as being planar terrain surfaces. At the end of the process, every LIDAR point is labelled as either terrain or non-terrain point.

Once the terrain information is extracted by the RTF filter, LIDAR points labelled as non-terrain features are further classified into more-detailed object classes, i.e., low-rise and high-rise objects. With a heuristic knowledge of the low-rise objects such as cars, bushes, and fences, outlying points with a height of less than a pre-defined threshold from the generated terrain surface are removed, and thus high-rise objects including buildings and trees are obtained. Then, differentiating building objects from trees is made by the use of IKONOS multi-spectral bands. The normalized difference vegetation indices (NDVI) are computed by a combination of red and near-infrared channels of IKONOS, by which the IKONOS image pixels are segmented into two groups, i.e., vegetated or non-vegetated pixels. As LIDAR points are projected onto the NDVI map, vegetated LIDAR points are detected and removed. As a result, LIDAR points which belong to building structures can be obtained. Further processing allows the individual objects to be bounded with a rectangle and these building "blob" polygons are then fed into the building extraction process with the final labels of LIDAR points.

The next step is to reconstruct building boundaries from each building blob polygon. This is achieved by a BUS (Building Unit Shape) organization method. It is assumed that generic building shape can be reconstructed by a set of convex polygons, and each

convex polygon is referred to as the BUS. A building blob polygon is used as an initial BUS. Line segments extracted from IKONOS imagery and LIDAR data are employed as hyperlines to halve that initial polygon. As a hyperline intersects the initial BUS, two sub-convex polygons are generated. This binary partitioning process continues until no child polygon is generated by the hyperlines, and ends up with a set of polygons (i.e., BUSes). This recursive generation of convex polygons is implemented by a Binary Space Partitioning (BSP) algorithm, in which a binary tree is incrementally expanded as child polygons are generated by the hyperlines. The final result of the BSP tree constructs a BUS space to represent a BUS adjacent graph where each node represents a BUS and each arc means the connectivity between neighbouring BUSes. A BUS grouping process merges only polygons, which belong to building structures, and eliminates spurious lines besides building boundaries. As a result, building outlines are reconstructed.

The hyperlines are generated in two different ways: data-driven and model driven methods. The data-driven hyperlines, generated as line segments, are extracted from IKONOS imagery by the Burns algorithm (Burns et al., 1986), and verified as building boundary lines by the support of the labelling result of LIDAR points. These boundary lines form a part of hyperlines for the generation of BUS space. However, due to the poor resolution of IKONOS imagery, line cues constructing building outlines may be completely lost or highly fragmented. To cope with this problem, the second constraint of the prismatic building model is assumed that generic building shape is made in some degree of geometric regularity. Based on this, the rest of hyperlines (i.e., model-driven hyperlines) are generated by employing simple building models. To this end, parallel lines and "U" structured lines are inferred over LIDAR space from each data-drive hyperline. It is, however, subjective as to the degree of complexity of individual buildings and to what percentage of the model-driven hyperlines must be used for reconstructing the building boundaries.

A balance between the data-driven and model-driven hyperlines is achieved by a level-of-detail (LoD) strategy. For constructing a BSP tree by the hyperlines, the most "significant" part of building structure, which comprises only building label points with a maximum size, is obtained first. Then, a BUS with less "significant" part is subsequently generated. In this way, the recursive segmentation of building parts

continues until only distracting polygons are generated by hyperlines. The LoD strategy prevents erroneous model-driven cues from being involved in building outline generation. Once the BUS space is generated, a heuristic filter verifies BUSes as building parts, which belongs to the building structure. Finally, the verified BUSes are aggregated, by which the boundary of polygonal building shape can be reconstructed.

1.4 Achievements

The ultimate goal of this study was to develop a unified framework for automatic extraction of buildings using monocular IKONOS satellite image and airborne LIDAR data. It was achieved through the extension of several existing algorithms and concepts reported by other researchers, as well as introducing novel methods, which are developed and implemented in this thesis. In this regard, the major achievements can be described as follows:

1. Introduced a hierarchical strategy to represent different urban features (terrain surfaces, low-rise objects, trees, and buildings) for automatic building detection. The original concept was adopted from Baillard & Maitre (1999), but each object class was uniquely represented in order to cope with a variety of object shapes by minimal use of geometric constraints.
2. Introduced a new concept for automatically reconstructing digital terrain models (DTMs) from a cloud of LIDAR points based on a "coarse-to-fine" regularization of terrain slope variations. This was implemented as a LIDAR filter, called recursive terrain fragmentation (RTF) filter.
3. Introduced a new concept for geometric modelling of generic building boundaries based on some fusion principles, in which complementary characteristics of IKONOS images and LIDAR data were integrated and data-driven and model-driven linear cues were optimally combined. This was implemented as a model generation tool, called building unit shape (BUS) organization.

All the proposed methods and algorithms were implemented by IDL (Interactive Data Language), which was an array oriented programming language developed by Research Systems, Incorporated (RSI). The performance of the developed techniques was evaluated using real data. Initial works of this study has been published in several

international conferences, which can be found in Sohn & Dowman (2001), Sohn & Dowman (2002), and Sohn & Dowman (2003). Also, the techniques resulting from this study has been employed and tested for carrying out ALFIE (Automatic Linear Feature Identification and Extraction) project in cooperation with QinetiQ and Laser-Scan Ltd., which has been sponsored by the Ministry of Defence of the United Kingdom.

1.5 Thesis outline

Chapter 1 introduces the problem statements, the research objectives of this study, the proposed method and strategy, and the structure of this thesis.

Chapter 2 reviews the fundamental theory of the IKONOS satellite, and airborne LIDAR systems. The history, systems, and applications of IKONOS and LIDAR are discussed. It also discusses previous research work reported in literature, which are related to LIDAR filtering and building reconstruction techniques.

Chapter 3 presents a new technique to reconstruct a terrain surface model from a point cloud of LIDAR measurements. Existing LIDAR filters are discussed with respect to the filtering mechanisms designed, and their pros and cons are described in relation to "terrain variation effect" and "object scale effect". The research leads to the development of the recursive terrain fragmentation (RTF) filter that enables more effective representation and reconstruction of various landforms with minimal use of parameters.

Chapter 4 evaluates the performance of the RTF filter using two full scenes of LIDAR data sets, the one in a suburban and the other in a complex urban area. The filtering result obtained from a suburban scene is compared to various reference datasets, with which the resulting filtering errors are analyzed. This result leads to the improvement of the RTF filter in order to detect very low objects and discontinuous terrains. The modified RTF filter is applied to a more complicated urban scene, and the filtering result is compared to manually filtered terrain surface.

Chapter 5 presents a method to extract building objects from monocular IKONOS image. Some difficulties of building extraction using single image are discussed with respect to the scene complexity and resolution effect. This analysis leads to the

development of a middle-level of cue grouping process, called BUS (Building Unit Shape) organization, which recovers highly fragmented or completely missed linear cues. The developed technique is applied to a sub-scene of IKONOS image, and the limitations of the method initiate the study of chapter 6.

Chapter 6 presents a method to automatically detect buildings and describe their boundaries from monocular IKONOS image and LIDAR data. A building detection process to isolate individual buildings from their surroundings is presented. Dominant object classes comprising an urban scene are hierarchically represented, and methods to detect each object are presented. An initial building shape is delineated with a rectangle obtained by the building detection process. A method for overcoming cue deficiency by integrating data-driven and model-drive linear cues is introduced. Considering the contribution of LIDAR data, the BUS organization method presented in chapter 5 is modified, in which methods to generate polygonal cues and to verify them are introduced. The technique is applied to a sub-scene of IKONOS image, and the building extraction results are compared to reference building vectors.

Chapter 7 summarizes the conclusions of the research and the suggestions for future study.

CHAPTER 2
BACKGROUND

2.1 Introduction

Chapter 1 introduced the subject of building extraction from remotely sensed data as a primary research topic of this study. The aim of this chapter is to present preliminary background information for the research works demonstrated in this thesis. In order to get a better understanding of characteristics of the primary data sets, this chapter begins with a description of the systems that acquired the data used in this thesis: the IKONOS satellite, and the airborne LIDAR. It then goes on to review LIDAR filtering and building extraction techniques in literature, which provided the initial motivation for this research.

2.2 IKONOS satellite system

2.2.1 Introduction

Since NASA initiated the Landsat series in 1972, a number of earth observing satellites such as SPOT, IRS, and MOMS have demonstrated their capability for providing repetitive earth resource information with medium resolution imagery. Metric accuracy of those medium resolution satellite imaging sensors has been, however, limited to the order of 30-60m for Landsat, 10-25m for SPOT, and 5-15m for MOMS and IRS. These accuracies are not attractive to the users when sub-metre level precision is so easily attainable from aerial photography. Due to this reason, the medium resolution satellites have been mainly used for remote sensing applications at small scale.

Launched in September of 1999, IKONOS satellite has initiated a new era of commercially available, high-resolution satellite imagery. From a 680 km sun synchronous orbit, the IKONOS satellite simultaneously collects 1-meter panchromatic and 4-meter multi-spectral images in 4 bands with 11-bit resolution. Accurate measurement of interior and exterior satellite orientation enables IKONOS to achieve high accurate geo-positioning accuracy with or without ground control points (GCPs). In addition to offering high-resolution and multi-spectral data, the IKONOS satellite provide a short revisit time (2.9 days) and the capability to perform stereo mapping. In particular, stereo pairs can be formed in near real time due to a very flexible pointing mechanism. These superior characteristics make the IKONOS imagery very attractive for a wide variety of applications, especially in the field of large scale topographic mapping.

In this thesis, IKONOS imagery is used as a data source for building extraction purpose. In §2.2.2 and §2.2.3, the overall system of the IKONOS satellite and its data acquisition properties are introduced. In §2.2.4, the geo-positioning accuracy of IKONOS imagery are discussed based on the literature. Finally, a number of applications using IKONOS imagery in literature is presented in §2.2.5.

2.2.2 System overview

IKONOS is operated in a near-polar obit with the inclination of 98° at an altitude of 680 km so the orbit is slightly slanted relative to a line running between the North and South poles. Following the near-polar orbit, IKONOS travels northwards on one side of the Earth (ascending pass) and then toward the southern pole on the second half of its orbit (descending pass). Figure 2-1 shows a near-polar orbiting satellite. The orbit is also sun-synchronous such that the position of the sun in the sky as the satellite passes overhead will be the same at a constant local solar time. In the sun-synchronized orbit, IKONOS in the ascending pass is on the shadowed side of the Earth while the descending pass is on the sunlit side. Thus, a passive sensor of IKONOS, recording reflected solar energy, only images the Earth's surface on a descending pass, which crosses the equator at a local solar time of 10:30 am. This ensures constant illumination conditions when acquiring images in a specific season over successive years, or over a particular area over a series of days. This is an important factor for monitoring changes between images or for mosaicking adjacent images together, or for stereo-matching multiple

images over the same area, as they do not have to be corrected for different illumination conditions.

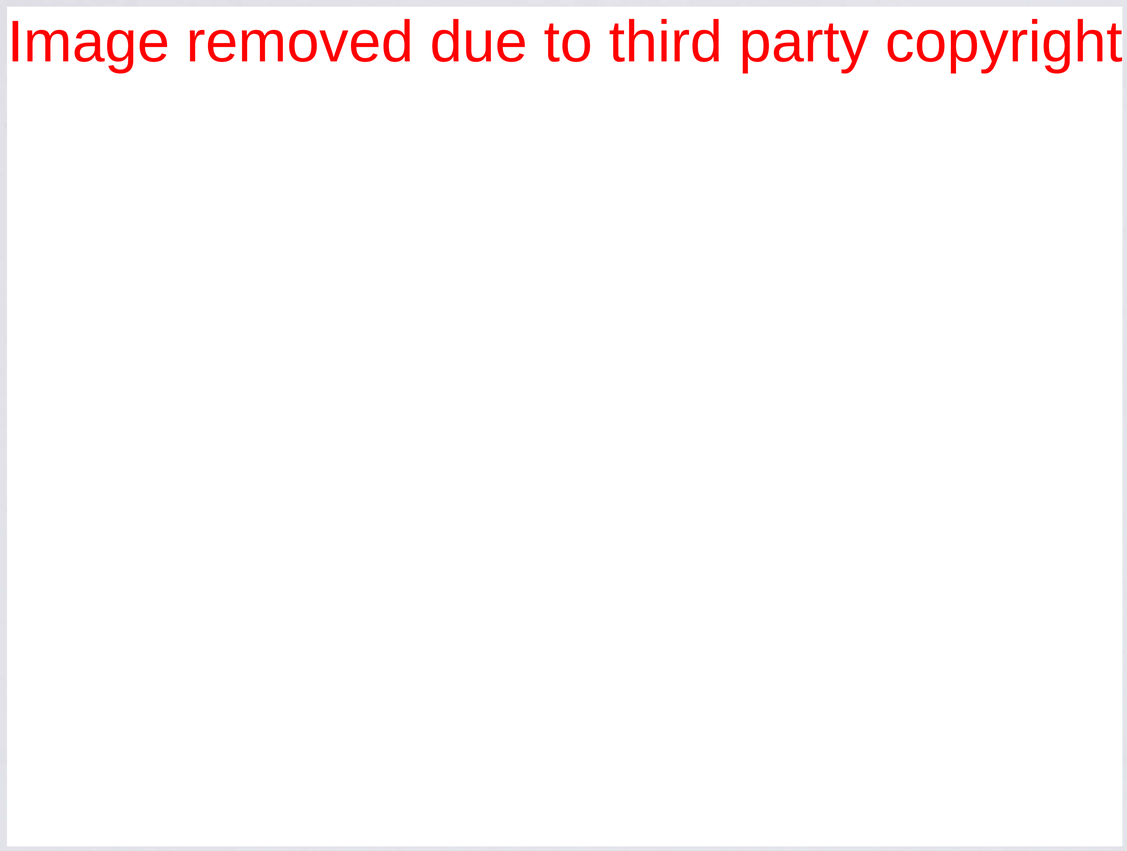


Figure 2-1. Illustration of a near-polar orbiting satellite

With the speed of about 7 km/s, IKONOS completes its orbit in a period of 98 minutes, which implies that IKONOS makes 14.6 revolutions a day. During each revolution, the Earth rotates approximately 24.5° so that IKONOS will observe different part of the Earth in the next orbit. In the sun-synchronous orbit, a repeat cycle of IKONOS is 14 days which means that the IKONOS passes vertically (i.e., at the nadir point) over the same location on the Earth's surface for a second time in 14 days. Unlike conventional remote sensing satellites such as Landsat, IKONOS can be directed to image Earth's surface with a range of diversified oblique viewing angles (i.e., forward, backward, left and right of the satellite position). Hence, the satellite can view the same geographical location at an off-nadir position, thus making the "revisit" time less than the repeat cycle. At latitude of 40°, an oblique viewing angle of 26° leads to the revisit time of 2.9 days, while 1.5 days will be taken for revisiting the same location with an oblique viewing angle of 44°. The revisit times are shorter for higher latitudes and longer for latitudes closer to the equator as the orbit paths come closer together near the poles. The revisit period is an important consideration for a number of monitoring applications, especially when frequent imaging is required. Table 2-1 shows the orbital parameters of IKONOS.

Table 2-1. IKONOS orbit parameters

Orbit Parameters	IKONOS
Orbit type	Near-polar, sun-synchronous
Orbit altitude	680 km,
Orbit period	98 min
Inclination	98°
Revolution per day	14.6/day
Repeat cycle	14 days
Revisit time	1-3 days

IKONOS was manufactured based on the LM-900 satellite bus designed by Lockheed Martin Co. The satellite was equipped with three solar panels and a sophisticated electro-optical imaging sensor. The dimension of main body is 1.8m (width) x 1.8m (length) x 1.6m (height) with the total weight of 726 kg. Figure 2-2 illustrates the overall IKONOS system.

Figure 2-2. Illustration of IKONOS system (Bayir, 2003)

The communication between IKONOS and the ground station can be established by two antenna systems; narrow-band antenna and wide-band antenna system. Narrow-band antenna system, called S-band antenna, is an omni antenna used for receiving the tasking packets from the ground station and transmitting the real-time response to the ground station. By this antenna system, the regional operation centres can control IKONOS by sending the imaging tasks to the satellite within 25 seconds after the first contact with IKONOS. Once the imaging task is received from the ground station, an operational "scenario" of IKONOS for the imaging mode is calculated by on-board software and sent to the subsystems and actuators to precisely position the satellite for accomplishing the required imaging task. Then, wide-band antenna, called X-band antenna, sends captured images in real-time mode to the ground station with the downlink speed of 320 megabit-per second.

While flying in space, the spatial positions of the satellite and sensor are collected from on-board GPS data. Then, the satellite ephemeris is determined by post-processing the GPS data on the ground using a sophisticated filtering and orbital modelling algorithms. The satellite attitude is rigorously determined by on-board star trackers and gyroscopes with different rates. The star tracker data is measured at lower frequency with high absolute accuracy, whereas the gyroscope data is measured at higher frequency with low absolute accuracy. A Kalman filtering post-processor coherently integrates both datasets and the overall satellite attitude data can be determined very accurately over short time interval. However, the star tracker makes errors of the attitude measurement when the star tracker is disturbed by the sun's light (Bayir, 2003). The relationship between the satellite attitude coordinate system and the IKONOS sensor coordinate system is described by the interlock angles. The initial interlock angles are determined by pre-launch assembly measurements and later refined by in-flight calibration (Grodecki & Dial, 2001). With the satellite attitude and ephemeris, the exterior orientation of the IKONOS imaging sensor can be determined.

The interior orientation of the sensor is described by the Field Angle Map (FAM). The FAM provides the optical distortion parameters and the focal plane array layout. By the FAM the line-of-sight vector for each image pixel can be determined in the sensor coordinate system. With the satellite attitude and interlock angles, the FAM allows one to compute the pointing direction of every image pixel. The interior orientation

parameters are calibrated in both pre-launch and in-flight phases. A more detail discussion of the calibration for the interior and exterior orientation of IKONOS can be found in Grodecki & Dial (2002).

The image acquisition of IKONOS is performed by the Optical Sensor Assembly (OSA), which was manufactured by a state-of-the-art technology of Eastman Kodak Co. (Kodak). The OSA comprises an optical telescope, panchromatic and multi-spectral imaging sensor arrays, and processing electronics. The optical telescope has the equivalent resolving power of a 10-meter focal length telescope. The telescope features three concave mirrors, each precisely configured to capture and focus high-resolution imagery onto the imaging sensors at the focal plane. Two additional flat mirrors "fold" the imagery across the inside of the telescope before coming to precise focus on the digital imaging sensors. This significantly reduces overall telescope length to 2 m without any loss of image quality and weight by almost a factor of two. Figure 2-3 shows a cross section through the OSA.

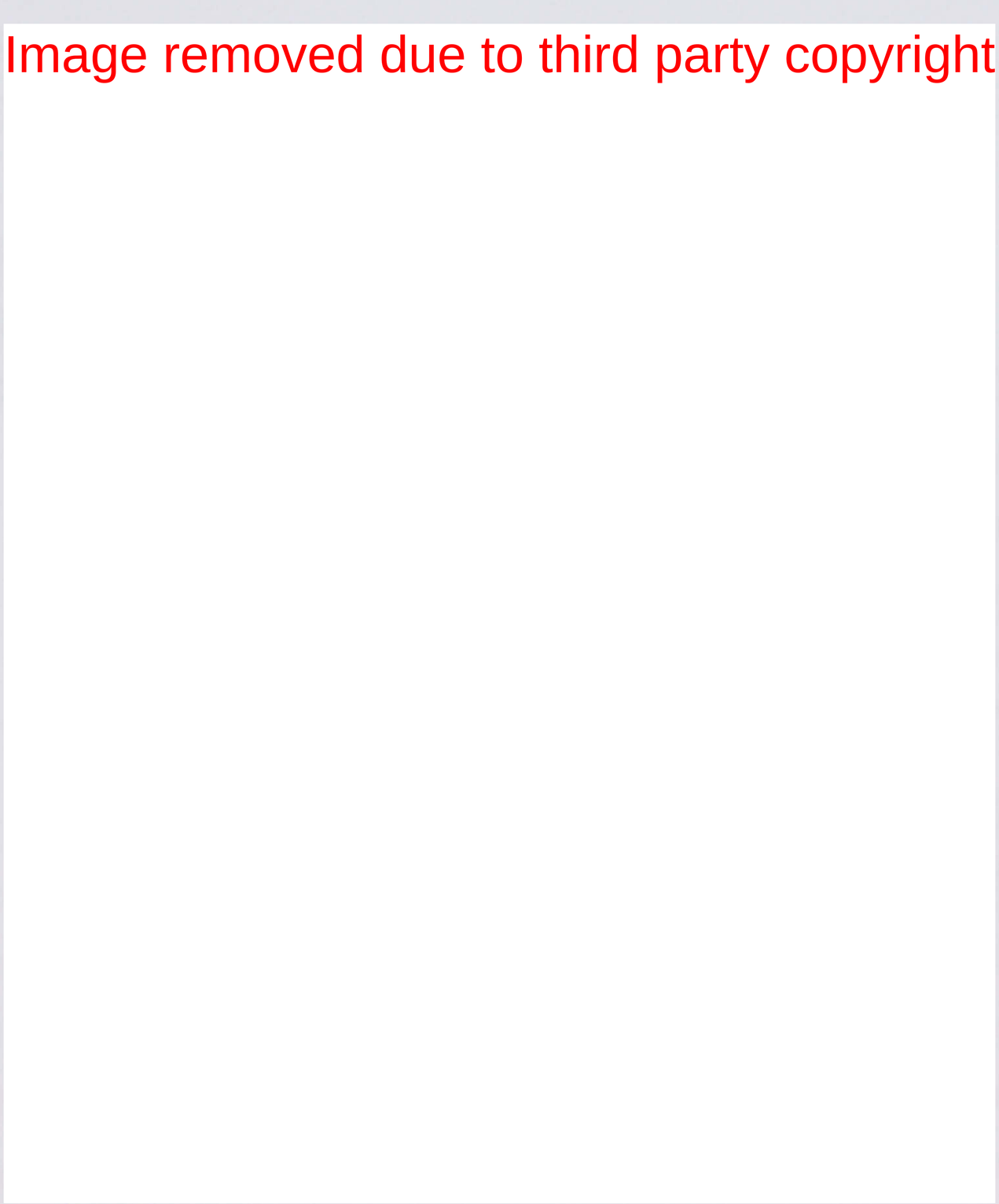


Figure 2-3. Cross section of the OSA

The light guided by the optical telescope is then simultaneously converted into panchromatic and multi-spectral imagery by CCD (Charge Coupled Device) sensor array. A special filter with five very thin layers stacks on the same piece of glass, and separates the components of blue (0.45-0.52 μm), green (0.52-0.60 μm), and red light (0.63-0.69 μm), as well as the near-infrared waves (0.76-0.90 μm). The fifth stack is applied to the flip side of the glass for a panchromatic band, which allows the panchromatic band (0.45-0.90μm) to pass through the mirror. The spectral resolutions of the multi-spectral bands are selected as being compatible to the Landsat's TM (Thematic Mapper) sensor. Table 2-2 presents the sensor parameters of IKONOS satellite.

Table 2-2. IKONOS sensor parameters

Orbit Parameters	IKONOS
Focal length	~10 m
Field of view	0.93°
Swath width	11 km
CCD array length & pixel size	13,800, 12 μm (PAN) 3,375, 48 μm (MS)
Spatial resolution	0.82 (PAN), 3.28 (MS)
Radiometric resolution	11 bits (PAN & MS)
Spectral resolution	0.45 - 0.90 microns (PAN) 0.45 - 0.52 microns (Blue) 0.52 - 0.60 microns (Green) 0.63 - 0.69 microns (Red) 0.76 - 0.90 microns (Near IR)

Solar energy passing through the filter is then captured by the linear CCD arrays. The linear array contains 13,500 CCD (Charge Coupled Device) elements with 12 μm pixel size for the panchromatic band. For the multi-spectral bands, 3,375 CCD elements with 48 μm pixel size are required for each spectral band. The four multi-spectral CCD arrays including the near-infrared band are placed on a single integrated array, rather than acquiring multi-spectral imagery across separate photo detectors. The CCD elements in panchromatic and multi-spectral line arrays are rigidly attached to the focal

plane in a stable thermal-mechanical environment. The input passing through each CCD element is quantized into 11 bits for all bands, representing a digital value of each pixel of the image recorded. The spatial resolution at nadir is 0.82 m and 3.28 m for the panchromatic and multi-spectral band respectively. The image data is compressed to a fraction of its original size for either storage on-board the satellite or real-time transmission to a ground station. The image compression algorithm developed by Kodak compresses 11-bits-per-pixel image to 2.6 bits per pixel.

2.2.3 Data acquisition

The image acquisition of IKONOS satellite is performed by a pushbroom sensor. The pushbroom sensor uses a linear array of CCD elements to generate an image of the ground line by line. As the satellite moves forward, the image of a strip of ground is projected onto the linear CCD array by the optical telescope. A bunch of photons "hit" to each CCD elements are accumulated within a certain pre-determined time interval (i.e., dwell time) and converted into an electrical output. The electrical output then turns into a digital value forming a pixel of digitally formatted image, which represents the magnitude of a reflecting solar energy for a single ground resolution cell (D) in figure 2-4. A physical dimension of the ground resolution cell is often referred as the ground sampling distance (GSD), which is the width of the ground resolution cell. GSD describes the spatial resolution of the satellite pushbroom sensor, which means the size of the smallest possible feature that can be detected. As the flying height of the satellite is almost fixed, GSD is determined only by the size and IFOV (Instantaneous Field of View) of the individual CCD element. The IFOV is the angular cone of visibility of each CCD element. As for IKONOS, the panchromatic sensor has 0.82 GSD at nadir, while the GSD of the multi-spectral sensor is four times that of panchromatic sensor, i.e., 3.28 m at nadir.

Another element characterizing a pushbroom sensor is the swath width which is the length of an image strip that can be taken by a linear CCD array. Like the computation of GSD, the swath width is determined by a total viewing angle of the sensor, called as FOV (Field of View). The FOV is determined as IFOVs are summed up across the total number of CCD elements comprising a pushbroom sensor. With the FOV of 0.93°, IKONOS has 11 km swath width. When the sensor has moved forward a sufficient distance the next strip of ground successively forms the next line of image produced. As

a result, a two dimensional image is acquired over a period of time. Figure 2-4 illustrates a typical pushbroom sensor in nominal operation.

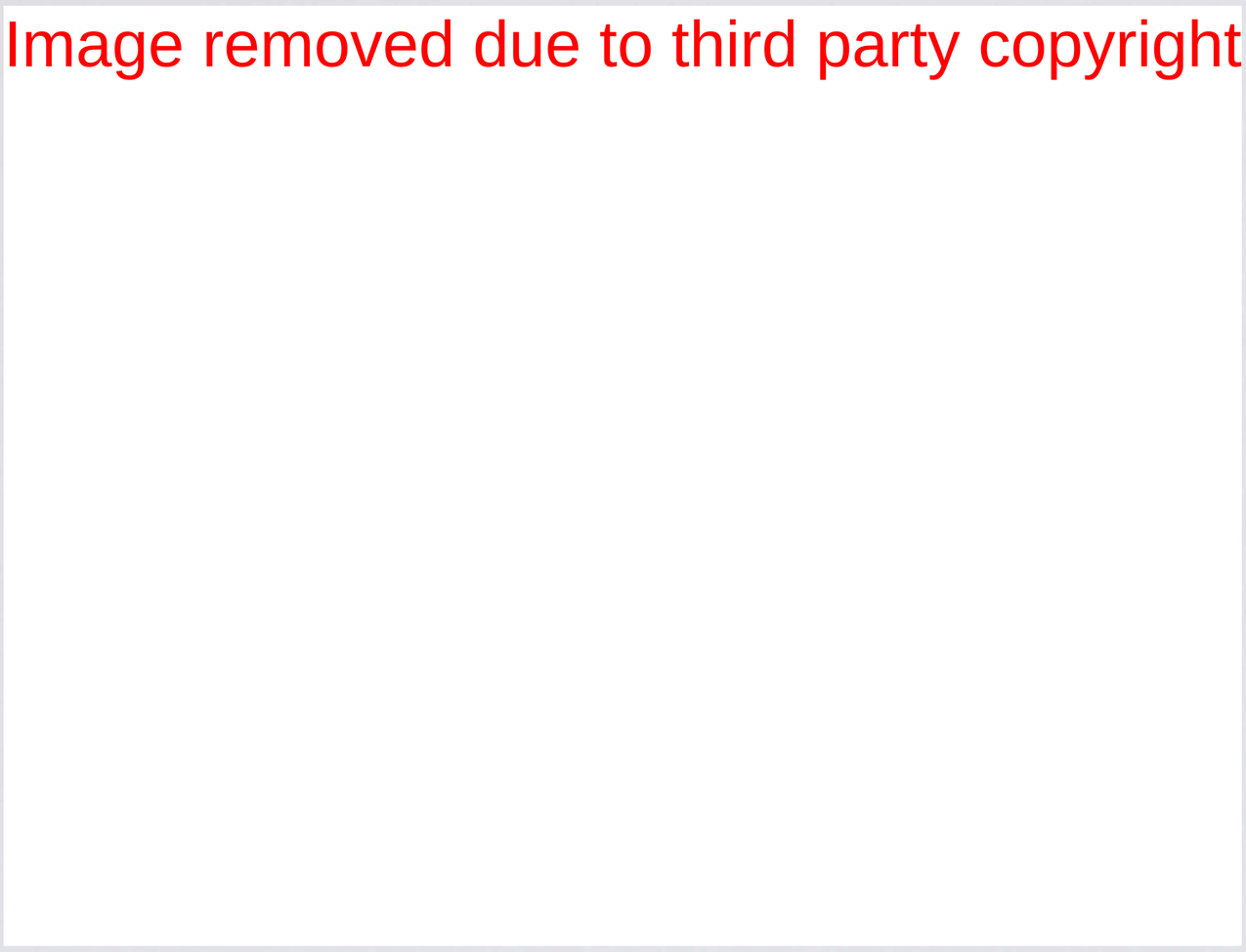


Figure 2-4. Typical pushbroom sensor operation

In general, a pushbroom sensor has a number of advantages over other electro-optical scanning devices. Comparing to a rotating mirror, a liner pushbroom sensor can increase the dwell time very efficiently which allows more energy to be detected by each CCD element. The increased dwell time also facilitates smaller IFOVs and narrower bandwidths for each detector. As a result, finer spatial and spectral resolution can be achieved without impacting radiometric resolution. In addition, the solid-state microelectronic devices used for the sensor are relatively small, lightweight, operate with less power and have no moving parts. All of these attributes are key elements for designing a compact and longstanding platform. For this reason, the pushbroom sensors have been employed on both airborne and spaceborne platforms. In particular, as for airborne pushbroom sensors producing very high spatial resolution, HRSC (High Resolution Stereo Camera) A/AX/AXW series of sensors and ADS40 are now available from DLR and LH System respectively. Also, most of high-resolution satellites such as QuickBird, Orbview-4, and EROS-1A including IKONOS have carried the pushbroom sensors.

One of the reasons for the IKONOS system being unique is the ability to rotate the satellite position to any angle to acquire an image to the side, forward, or backward of the ground track. As a whole, the satellite can be rotated up to ±45° in both along and across flight direction. With this powerful off-nadir viewing capability, more flexible imaging tasks can be accomplished. IKONOS mono image strip is mainly acquired in along track scanning mode with a range of off-nadir viewing angles. The length of IKONOS mono image strip typically varies from 11 km to some 200 km. The off-nadir viewing increases the distance between the sensor and target increases, which leads to the change of the GSD. The GSD of 0.82 m for panchromatic images and 3.28 m for multi-spectral images at nadir will be increased to 1 m and 4 m respectively at 26° off-nadir. As the GSD increases, the swath width is also changed. For instance, the swath width of 11 km at nadir will be increased to 13.8 km at 26° off-nadir.

Figure 2-5. IKONOS mono and stereo image collection (Dial & Grodecki, 2003)

Unlike SPOT, which takes across-track stereo images from different orbit passes, IKONOS collects along-track stereo pairs in the same orbit. Figure 2-5 shows a typical stereoscopic collection operation of IKONOS satellite. As the satellite approaches the target, it is manoeuvred to collect the first image of the stereo pair while pointing in a forward direction. A hundred or so seconds later, the satellite is manoeuvred to again the same area to collect the second image, this time pointing in a backwards direction.

This forward and backward collection of overlapping swaths provides optimal base height ratios for a stereo scene, plus opportunities for additional imaging during the same time frame. In addition, the along-track stereo pairs are advantageous for subsequent processing such as DEM generation due to the identical scene content and illumination conditions for both two images.

The collected image data are transmitted in real-time to ground via the wide-band antenna. After all imagery of the pass has been received at the ground station, a metadata for collected imagery is transmitted. The metadata records the image acquisition geometry of IKONOS satellite, which is described by the azimuth and elevation angles from the image areas to the satellite. The sensor elevation angle is the angle from the horizontal plane up to the satellite. The projection of the line of sight from the satellite onto the horizontal plane defines the sensor azimuth. Sensor azimuth is measured clockwise from North. In the metadata, the azimuth and elevation angles to the sun are also contained. Figure 2-6 illustrates the imaging geometry of the IKONOS satellite.

Figure 2-6. IKONOS imaging geometry (Grodecki & Dial, 2001)

2.2.4 Geo-positioning accuracy

A two-dimensional image acquired by the pushbroom sensor is described in an image space coordinate system (i.e., line and sample). For mapping purpose, the image coordinates must be geo-positioned so that the location of each image pixel on the ground can be represented in a map projection intended for the application (i.e., object space coordinate system). For this purpose, the functional relationships between the image space and the object space must be known. The best known method to geo-position the satellite imagery is accomplished by the support of a physical sensor model. The physical sensor model provides the exterior orientation describing the position and orientation of a sensor with respect to an object space coordinate system, as well as the interior orientation describing the optical and electrical design parameter of the sensor. Once the sensor model is established, the image-to-object transformation is performed by a collinearity equation, which defines a linear relationship between the image and object spaces. The physical sensor model is rigorous and its parameters are statistically uncorrelated, as each parameter has a physical significance. Most of the pushbroom sensors have been parameterized by the physical sensor model. However, obtaining the rigorous precision of the sensor modelling requires a large amount of sensor related information, such as orbit parameters, sensor platforms, focal length, ephemeris data, relief displacement, Earth curvature, atmospheric refraction, lens distortion, etc. Although it has been well established in today's digital photogrammetric workstations (DPWs), the modelling process is normally time consuming and different types of sensor need different models.

The physical sensor models are not always available, especially for the case of IKONOS imagery. Space Imaging, which is the data vendor of IKONOS, has adopted the rational function model (RFM) as a replacement sensor model for image exploitation. IKONOS imagery is supplied with the parameters of RFM instead of providing a physical sensor model. RFM provides a means of reparameterising the sensor interior and exterior orientation without explicit reference to either a sensor model or satellite ephemeris. Thus, the users cannot access to the proprietary sensor model and precise ephemeris data for the IKONOS satellite imaging system. The RFM, often called rubber-sheet model, is essentially a generic form of polynomials. It defines the formulation between a ground point and the corresponding image point as ratios of two polynomials to approximate the rigid imaging geometry. As discussed in (Dowman

& Tao, 2002; Tao & Hu, 2001a), RFM can describe the object-to-image space transformation in "forward" or "inverse" form, and it can be "terrain dependent" or "terrain independent". As for the IKONOS imagery, the RFM is provided by Space Imaging in "forward" and "terrain independent" form, which means that the sensor model describes the object-to-image space transformation, rather than the image-to-object space transformation and the ground truth is not associated with the model generation.

A general formulation of the terrain independent and forward RFM can be described as (Fraser et al., 2002a; Grodecki & Dial, 2001):

$$x_n = \frac{P_1(X,Y,Z)}{P_2(X,Y,Z)}, \quad y_n = \frac{P_3(X,Y,Z)}{P_4(X,Y,Z)} \tag{2-1}$$

where, x_n and y_n are the normalized (offset and scaled) image coordinates (line, sample) for the image and X,Y,Z are the corresponding object point coordinates, which are referred to normalized latitude, longitude and height. In this model, each polynomial, P_i, where $1 \le i \le 4$, has the general polynomial from, which is given by

$$P(X,Y,Z) = \sum_{i=0}^{m_1}\sum_{j=0}^{m_2}\sum_{k=0}^{m_3} a_{ijk} X^i Y^j Z^k \tag{2-2}$$

In normal use, the order of the polynomials is limited to $0 \le m_1 \le 3$, $0 \le m_2 \le 3$, $0 \le m_3 \le 3$, and $m_1 + m_2 + m_3 = 3$. Then, P_i is determined by a third-order polynomial with 20 rational polynomial coefficients (RPCs) as follows:

$$
\begin{aligned}
P(X,Y,Z) = {} & a_0 + a_1 X + a_2 Y + a_3 Z + a_4 X^2 + a_5 XY + a_6 XZ + a_7 Y^2 \\
& + a_8 YZ + a_9 Z^2 + a_{10} X^3 + a_{11} X^2 Y + a_{12} X^2 Z + a_{13} XY^2 \\
& + a_{14} XYZ + a_{15} XZ^2 + a_{16} Y^3 + a_{17} Y^2 Z + a_{18} YZ^2 + a_{19} Z^3
\end{aligned} \tag{2-3}
$$

To perform the object-to-image space transformation, four polynomials with 80 RPCs must be determined per image. As those polynomials are associated with Eq. 2-1, distortions caused by optical projection can be corrected by ratios of first-order terms, and corrections to Earth curvature, atmospheric refraction, and lens distortion, etc., can be well approximated by second-order terms. Some other unknown distortions with high-order components can be modelled with third-order terms (Tao & Hu, 2001b). Thus, a relationship between the object points and the conjugate image points can be obtained by RFM sensor model. However, the object-to-image space transformation has been described in the normalized space by Eq. 2-1. As additional parameters of 10 offset and scale terms are given, real image coordinate of (x, y) and the corresponding geographical coordinate of (ϕ, λ, h) can be computed by

$$
x = x_{scale} \times x_n + x_{off}; \quad y = y_{scale} \times y_n + y_{off} \tag{2-4}
$$

$$
\phi = \phi_{scale} \times X + \phi_{off}; \ \lambda = \lambda_{scale} \times Y + \lambda_{off}; \ h = h_{scale} \times Z + h_{off} \tag{2-5}
$$

where (x_{off}, y_{off}) and $(\phi_{off}, \lambda_{off}, h_{off})$ are the offset parameters; (x_{scale}, y_{scale}) and $(\phi_{scale}, \lambda_{scale}, h_{scale})$ are the scaling factors. The RFM becomes attractive due to its independence from the sensor platform and inexpensive computing cost. The method requires no knowledge of the sensor geometry. Thus, different sensor types can be easy to model within DPWs, with only changing RPCs. For this reason, the RFM solution has been recognized as a suitable approach for the military or public applications, which require a faster implementation of the real-time loop in DPWs (Dowman & Dolloff, 2000).

The RPCs provided with IKONOS imagery are calculated by Space Imaging. A least-square method is utilized to estimate the RPCs based on a 3D grid of virtual control points in the object space generated from a physical sensor model. The 3D grid of objects points can be generated as rays emanating from a 2D grid of image points are computed by the physical IKONOS sensor model and then intersected with a number of constant elevation planes defined in the object space (Grodecki & Dial, 2001). It has been shown by Grodecki (2001) that RPCs derived by Space Imaging performs the rigorous object-to-image space transformation to within 0.05 pixel accuracy when compared to the physical sensor model of IKONOS. With these high-quality RPCs, the image-to-object space transformation of IKONOS imagery is performed. This gives an accuracy in the 3D object space which is consistent with the specifications for the particular class of IKONOS products; Geo, Reference, Pro, Precision and Precision Plus (see Table 2-3). There are substantial differences of the achievable geo-positional accuracy across the products. It depends on availability of GCPs and terrain model and their quality. The RPCs for the Geo product are derived solely from satellite ephemeris and attitude data, which are expected to produce a RMS (Root Mean Square) positioning accuracy of about 25m. However, the RPCs for Precision level products are computed with the additional aids of ground control and high-quality terrain elevation model. Except for the Geo product, all the others are orthorectified images. In particular, Space Imaging has provided the RPCs with IKONOS imagery only for the Geo Ortho Kit and stereo products.

Table 2-3. IKONOS product specification (Space Imaging, 2003)

As mentioned above, the vendor-provided RPCs can approximate the physical sensor model for IKONOS satellite. However, these coefficients are derived from virtual control points, which are not usually associated with the ground truth. Thus, the RPC computed with no GCPs may cause relatively large errors in the geo-positioning process. For this reason, an intensive research has focused on improving the geo-positional accuracy of RPC-based IKONOS products. Hanley & Fraser (2001) reported that the IKONOS Geo imagery provides a strong metric integrity, which means that most geo-positioning errors remain as systematic biases with almost the same direction and magnitude. To remove those biases, the image-to-object transformation was done by a number of 2D transformations. First the control points are projected onto a reference plane, to reduce the effect of terrain relief, and then transform the image to these points using similarity, affine and projective transformations. With 6 GCPs and 20-25 check points, the results show that RMS geo-positioning accuracy of 0.3-0.5 m was achieved without using RPC solution. Fraser et al. (2002b) have extended this work for improving 3D positioning accuracy using an affine or DLT solution. With 4-6 GCPs, accuracy of about 0.5m can be achieved in X and Y and less than 1m in Z for both stereo and multi-image configurations. The affine projection method was considered as more practical approach compared to the DLT solution. This empirical affine sensor orientation model has been tested for the purpose of block adjustment, over a range of scene sizes and terrain types (Fraser & Yamakawa, 2003; Hanley et al., 2002). Both results showed that the affine model can produce pixel-level and even sub-pixel level geo-positioning accuracy for the Geo imagery without using the vendor-supplied RPCs.

The geo-positioning accuracy can be also improved as the vendor-provided RPCs are directly refined with the support of GCPs. Hu & Tao (2001) suggested a sequential adjustment method to update RPCs using additional GCPs. Fraser et al. (2002b) developed a bias compensation model that comprises one offset parameter per image coordinate (i.e., line and sample offsets). These parameters are estimated by an indirect least-square method with the provision of GCPs. The role of GCPs is to determine an image coordinate translation and thus correct the shift effects of the sensor platform. Then, the determined bias parameters are incorporated into the vendor-supplied RPCs, which results in the bias-corrected RPCs. The result showed that a single GCP is sufficient to generate 3D co-ordinates with a high accuracy less than 1 m. It was reported that the quality of the GCPs used is more important, than a number of GCPs

used or their configuration for improving the RPC model. Hanley et al. (2002) have extended this bias compensation model including drift effects of the sensor platforms in the along- and across-track direction. However, the result showed that the sensor drift correction does not improve the geo-positional accuracy. It confirmed the result of Dial & Gredecki (2002) that the sensor drift effect is not considered as being significant in the exterior sensor modelling for IKONOS, until the Geo strip length is at least 50 km long. A similar conclusion can be found in Dial & Grodecki (2002). The bias-corrected RPC model can be applied to either single image with known height, or multi-image networks for the block adjustment. Compared to the aforementioned empirical affine method, the bias-corrected RPC model requires much fewer GCPs, which can reduce the cost of establishing and measuring the GCPs and carrying out the affine model computation. However, the vendor-supplied RPCs must be provided in order to implement the bias-corrected RPC model.

2.2.5 Application

The potential to utilize IKONOS imagery has been investigated by a number of researchers since the satellite was launched. A recently published OEEPE (European Organization for Photogrammetric Research) report (Holland et al., 2002) describes research into the use of IKONOS data for a range of application by several European mapping agencies and institutions. Marshall (2002) evaluated the IKONOS Geo panchromatic imagery for updating topographic mapping in Ordnance Survey. A range of linear features were manually extracted from rural and urban scene. The resulting features were compared to the 1:25,000 map, provided by the Swiss Federal Office of Topography. The result concluded that some linear features such as major roads, railway tracks could confidently be extracted. Buildings were easy to detect and capture from IKONOS imagery. However, only a general outline could be detected and it was difficult to capture the detailed divisions of housings. The major drawback of 1m imagery was described as the difficulty in interpreting small linear features such as fences and paths. Although the mapping specifications varies from country to country, the result indicated that there is potential in topographic mapping, between 1:10,000 and 1:25,000, and IKONOS panchromatic imagery could be used for revision of major topographic features at these scales. IKONOS imagery was found to be a viable alternative to aerial photography for checking the current mapping quality. This was

helped greatly by the ease of ortho-rectification and the large area coverage provided by satellite imagery.

Ryden & Sjöhed (2002) reported on the evaluation of the IKONOS 4m multi-spectral imagery against the specification for the Swedish 1:10 000 scale map. Two different band combinations (i.e., RGB and CIR combination) were used for manually evaluating four different topographical themes; built-up areas, road network, land cover, and drainage. In built-up areas, aerial features of industrial and residential buildings were suitable for updating the map. However, it was not enough to outline the details of small houses. For linear features, major roads and railways were clearly visible and even minor road could be identified in the countryside. The infrared band combination applied in forested areas revealed that different tree types and compositions could be discriminated. The infrared band combination was also very useful to detect and identify open water, rivers and streams. It was concluded that the multi-spectral IKONOS data is suitable to use for update of most features in the map, satisfying the specifications of the Swedish 1:10 000 map.

A number of algorithms have been developed to aid a topographic mapping process using IKONOS images. The algorithms were designed to automatically extract features such as buildings, roads, trees and land cover outlines from IKONOS imagery. Baltsavias et al. (2001) and Fraser et al. (2001) presented results of building extraction from IKONOS stereo images. Buildings were manually extracted with an in-house developed software tool and modelled in 3D space using the software package CC-modeler. A comparative study with the results obtained from airborne photographs concluded that about 15 % of the building area as measured in aerial images cannot be modelled. The shortcomings in radiometric quality both within and between IKONOS scenes were addressed for IKONOS imagery to give rise to problems of accuracy and interpretability in manual building extraction. Lee et al. (2003) showed an automatic building extraction algorithm from IKONOS Geo image. The study aimed to obtain approximate locations and shapes of potential buildings based on the spectral characteristics of multi-spectral bands. The ECHO classifier and ISODATA algorithm were used for supervised and unsupervised classification respectively. Based on the classification results, a Hough transformation-based building squaring algorithm extracted rectilinear building boundaries. The result demonstrated 64.4 % building

detection rate at about one pixel edge alignment accuracy. Muller et al. (2001) combines stereo images and multi-spectral bands. A digital surface model (DSM) was generated using the Socet Set software, and then a bare-earth surface was extracted. With the bare-earth surface produced, the normalized difference vegetation index (NDVI) served as differentiating buildings from trees. The method is limited to only detect building regions of interest with no boundary extraction.

Dial et al. (2001) developed an algorithm to extract road features from IKONOS imagery. IKONOS multi-spectral imagery was segmented into pre-determined feature classes by the classic Tasseled Cap algorithm. Based on that, road regions were roughly detected. An angular texture signature was measured for each pixel of the road regions, and classified depending on the number of the directions of the road about that pixel. Similar road regions were aggregated and a road topology was locally constructed. They showed that existing road network can be successfully matched to the road topology extracted. Gibson (2003) extended this technique to automatically extract a road network without use of existing vector. Besides the texture measurement and colour segmentation, the method considered linear features extracted by a modified Nevatia-Babu edge detector. The three different properties were independently measured and combined to characterize a road region. The entire image space was decomposed into similar road regions by a Generalized Balanced Ternary (GBT), which is a hexagon-based, multi-resolution representation of the image data. The decomposed road regions are topologically networked and converted into a graph structure. As a rule-based process verifies the road topology constructed, a road network was extracted and vectorized. Hu & Tao (2003) showed a perceptual organization technique to extract road centrelines from IKONOS panchromatic imagery. Linear features were extracted from a low resolution image pyramid, and those lines were then grouped in a hierarchical way, where long straight lines are grouped and short and curved lines are sequentially grouped to construct a road network. A pre-determined topology rule eliminated distracting short and isolated lines. The delineation of central lines was accomplished by a least-square template matching in an original resolution image pyramid.

The traditional multi-spectral classification method is not directly applicable to urban land use mapping at 1 m resolution because each land use class may be composed of

spectrally distinct land cover types. Zhang & Wang (2003) developed a rule-based land use inferring method using pan-sharpened IKONOS imagery. An unsupervised ISODATA algorithm is applied to produce spectral land cover classes. The rules to recognize land use from land cover are formulated from 36 typical land use samples. The rules are based on the density of different land covers in a neighbourhood. The results demonstrated that level II and level III urban land use classes of the USGS land use classification system can be recognized from IKONOS imagery with 88.50 % overall accuracy. Haverkamp & Poulsen (2003) developed an algorithm to detect urban changes from two IKONOS ortho-rectified images, consisting of both the panchromatic and multi-spectral bands. The urban changes of interest included new roads, repaved roads, distributed soil related to construction activities, and new buildings and building additions. The algorithm requires a number of necessary pre-processing steps to register and calibrate the images. Then, common "false" change detections such as caused by clouds, cloud shadows, and atmospheric differences are suppressed. The change detection process starts with a pixel-by-pixel comparison followed by change region analysis, verification, and labelling. For the change region analysis, the representative feature vectors were pre-determined using a Learning Vector Quantization (LVQ) algorithm for urban changes and suburban changes respectively. The Tasseled Cap Transformation (TCT) vectors derived from the multi-spectral bands are used for classifying the detected changes into pre-determined change categories. The result demonstrated that average accuracy of 68% is achievable.

2.3 Airborne LIDAR system

2.3.1 Introduction

A large number of x,y,z points are required to obtain an accurate representation of natural or man-made objects in 3D space. In the past, this detailed high-accuracy mapping has relies on an expensive method, requiring many man-hours of field surveying or work using photogrammetric soft-copy techniques. More recently, the stereo image-matching technique (Baillard & Dissard, 2000; Cord et al., 1999; Paparoditis et al., 2001) has made progress in automating such laborious work. Although this method provides good results for open terrain, problems occur in forest and build-up areas, where many geometric features are similar to each other, or homogeneous areas are frequent. Thus, the stereo image-matching technique has not

been matured yet to provide a "complete" solution for generating digital elevation models.

Since the mid 1990s, airborne laser scanners were developed as an alternative technique of traditional aerial photogrammetry and used for direct measurement of the topographical terrain surface. This relatively new technology, called LIDAR (Light Detection and Ranging), is an active sensing system that uses laser light to measure distances. When this system is mounted in a fixed wing or rotary wing airborne platform, distances between the sensor on the platform and points on the ground or non-ground objects are rapidly measured and generates densely spaced elevation data with a point density of up to 25 points per square meter. As the differential global positioning system (DGPS) and the inertial measurement unit (IMU) are integrated with the distances measured, LIDAR can provide a directly geo-referenced elevation data with a vertical accuracy of 15 cm and horizontal accuracy within 1/1000th of the flight height. Due to this nature of the LIDAR technology, numerous ground surveying points generating digital terrain models are not necessary. Also, high quality of three-dimensional visibility achieved by LIDAR can help the operator or computer-visional system easily recognize trees and buildings, even electrical transmission lines. For these reasons, LIDAR is becoming a major method for large-scale acquisition of elevation data.

The use of laser technology to measure distance between two separate objects, called laser ranging or laser altimetry, has been around for a long time since Schawlow and Charles established the theory of the optical laser in 1958 (Fowler, 2001). The term "laser" is an acronym of Light Amplification by Stimulated Emission of Radiation. This laser technology as remote sensing instrument has been used for more than 30 years. In 1964, laser ranging to a near-Earth satellite was initiated by NASA with the launch of the Beacon-B satellite. In the 1970s the laser remote sensing has been developed in the fields of lunar laser raging, satellite laser ranging, atmospheric monitoring and oceanographic researches. During 1980s the technology has been successfully demonstrated from a variety of airborne platforms such as NASA's Atmospheric Oceanographic Lidar (AOL) and Airborne Topographic Mapper (ATM) (Flood, 2001). In the 1990s, with the development of On-The-Fly (OTF) GPS techniques, small inexpensive IMU systems, and portable computing system, it became possible to

commercialize the technology and a wide range of LIDAR sensors mounted in airborne platforms is now available for commercial applications. A review of the commercial sector shows a rapid growth of LIDAR vendors which has grown in the last 5 years from 3 in 1995 to 58 in 2000 worldwide (Fowler, 2001) and it was predicted that 150~200 LIDAR sensors will be installed by 2005 (Flood, 2001).

LIDAR systems can be classified into two different types depending on the major spectral frequency used. A topographic LIDAR system working over land operates in the infrared portion of the electromagnetic spectrum, which is partially absorbed by water resulting in almost no returns, while a bathymetric system working over water uses the blue-green portion of the spectrum, where the blue laser pulse is reflected off the water's surface, and the green one can penetrates to the sea floor; the differentiated distance provides the bathymetric depth.

LIDAR can be also classified as either pulsed ranging system or continuous wave (CW) ranging system depending on the principle of the distance measurement (Ackermann, 1999). A pulsed ranging system is to measure the distance between a target and the sensor by the time difference of the transmitted and received laser pulses. The returned signal yields a narrow footprint on the surface, which can partly penetrate through the foliage to the ground. The pulsed ranging system can record multiple returns (currently up to five returns) from one instance pulse transmitted, which means that a part of beam on a tree is reflected from the treetops, but subsequent returns provides elevations of intermediate branches, and the last returning pulse would provide the elevation of the ground underneath the tree. The CW ranging system transmits a continuous stream of signals and digitizes the entire structure of returning waveform. A complete elevation profile within a wider footprint is captured by phase difference measurements between transmitted and received signals. In this thesis, the only relevant data product is LIDAR data acquired by the topographic system based on the pulsed ranging principle, which is the most popular LIDAR system. Table 2-4 shows parameters characterizing a typical topographic LIDAR system. This section deals with the overall technology of the airborne topographic, pulsed ranging LIDAR system.

Table 2-4. Characteristics of typical topographic LIDAR system (Fowler, 2001)

Image removed due to third party copyright

2.3.2 System operation

A LIDAR sensor mounted in an airborne platform emits rapid pulses of laser light, which are deflected toward the ground across the flight path by a scanning system. The transmitted laser pulse is reflected off the target on the ground and relayed to a light sensitive receiver. With a very accurate timing system, the round-trip travel time of a laser pulse between the transmitted and received signal can be determined. From this information, a relative distance between the laser sensor and the target is calculated. Since the distance is measured from the moving aircraft, the precise location of the laser sensor and attitude of the aircraft must be accurately known to correctly locate the reflected target in 3D space. To register the spatial position of the aircraft, a kinematic airborne GPS (Global Positioning System) receiver locks on to at least four GPS satellites. An inertial measurement unit (IMU) records the roll, pitch and heading of the

aircraft to measure the precise beam orientation at the time the laser pulse is transmitted. During the LIDAR surveying, one or more GPS ground reference stations are used to provide a DGPS solution for geo-referencing the received laser pulses to a local map projection. In addition to the geo-referenced raw data, some LIDAR systems provide intensity returns of reflected pulses, where objects with high reflectivity, such as a metal roof are encoded with higher gray values, while objects such as a newly paved black tarred roadway show lower gray values. With the laser scanning, many LIDAR systems carry a digital camera to take an aerial photograph and/or a video camera for reviewing scanned areas, which are used as auxiliary information for editing or classifying the LIDAR points collected. Figure 2-7 depicts a generalized schematic of airborne LIDAR mapping in operation.

Figure 2-7. Illustration of airborne LIDAR mapping in operation (Flood & Gutelius, 1997)

2.3.3 LIDAR system components

Figure 2-8 illustrates how LIDAR system acquires a geo-referenced point based on the measurement of two vectors. First is the measurement of the position, P_S, of LIDAR sensor with respect to the origin of a local mapping coordinate system, P_O, which results in a vector, $\mathbf{p}$. Second is the distance and the direction from the sensor, P_S, to a

reflected point, P_R, which is represented as a vector, **a**. Adding both two vectors of **p** and **a** leads to the vector, **r**, from P_O to P_R. As these two vectors are measured with high-accuracy for a massive size of transmitted laser pulses, a densely spaced digital elevation model can be generated by LIDAR surveying.

Figure 2-8. Geo-referenced point acquisition based on the measurement of two vectors (Katzenbeisser, 2003)

To measure these two vectors, most LIDAR systems are composed of several basic components as shown in Figure 2-9 Laser ranging and scanning systems contribute to measure the distance of $|\mathbf{a}|$ and the instantaneous viewing direction of **a** respectively. The positioning and orientation system (POS), which is integrated with DGPS and IMU, plays a role to measure **p**. In following section, each component is described in more detail.

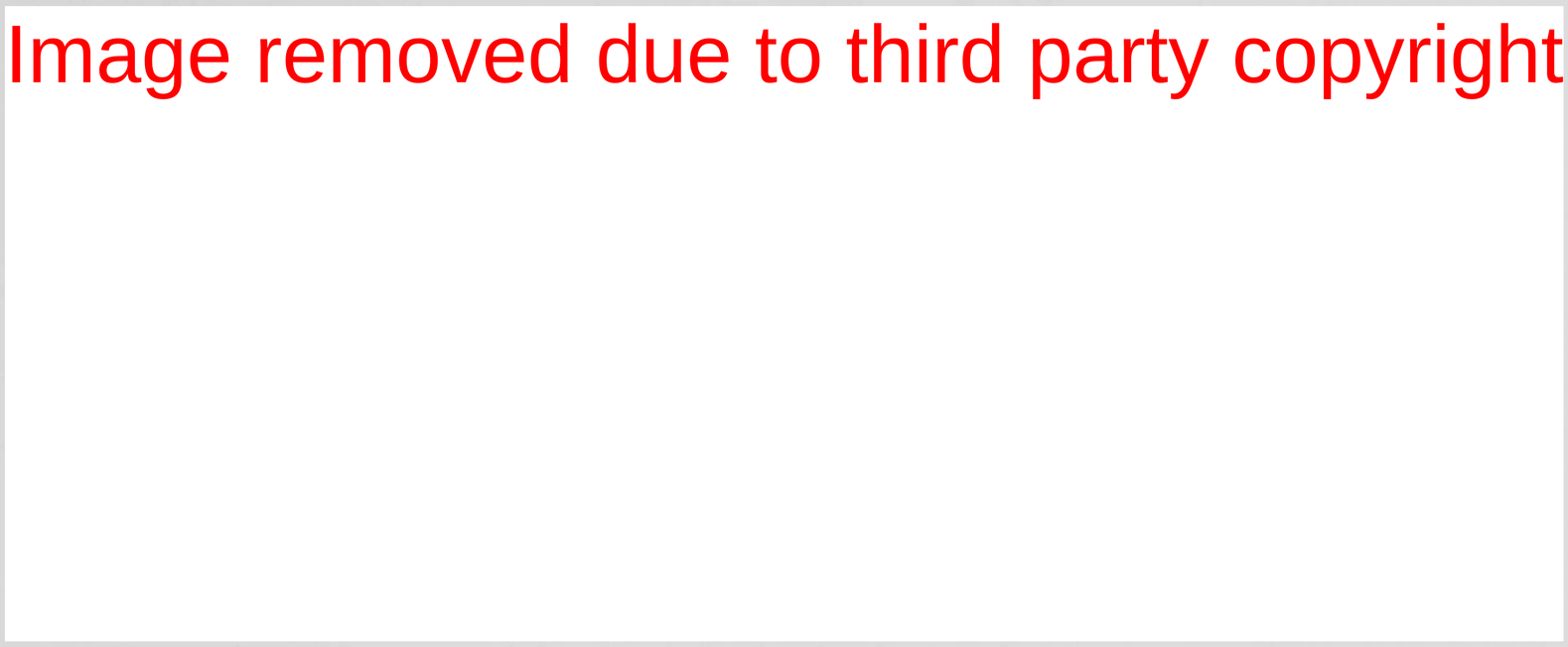


Figure 2-9. POS and Laser Sensor (Katzenbeisser, 2003)

Laser ranging

Laser ranging system comprises three components; laser transmitter, receiver, and time counter. The system measure a range from the sensor to the target on the ground. The laser transmitter mounted on an aircraft generates a number of bursts of light per second that is normally measured in kilohertz, kHz, commonly referred to PRF (Pulse Repetition Rate). A typical LIDAR system can generates 5,000 to 33,000 laser pulses in one second. Each discrete laser pulse is generated in certain time length, i.e., a pulse duration (or pulse width), which is the interval between the time, during the first transition, that the pulse amplitude reaches a specific fraction of its maximum amplitude, and the time the pulse amplitude drops, on the last transition, to the same level. Most LIDAR systems generate a discrete laser pulse with the pulse duration of 10 ~15 nanoseconds (ns).

The laser pulse generated is transmitted to the ground through a specific scanning system that deflects the direction of laser lights. With the pulse traveling at the speed of light, the receiver captures the returning echo reflected off an object on the ground. To avoid an ambiguity discriminating consecutive returning pulses, no pulse is transmitted until the echo of the previous pulse has been received. The elapsed time between two consecutive transmitting pulses is called as PRT (Pulse Repetition Time), which is the reciprocal of PRF, i.e., 1 PRT = 1/PRF), and PRF (or PRT) is independent of the pulse duration. Figure 2-10 depicts a generalized schematic of laser raging system.

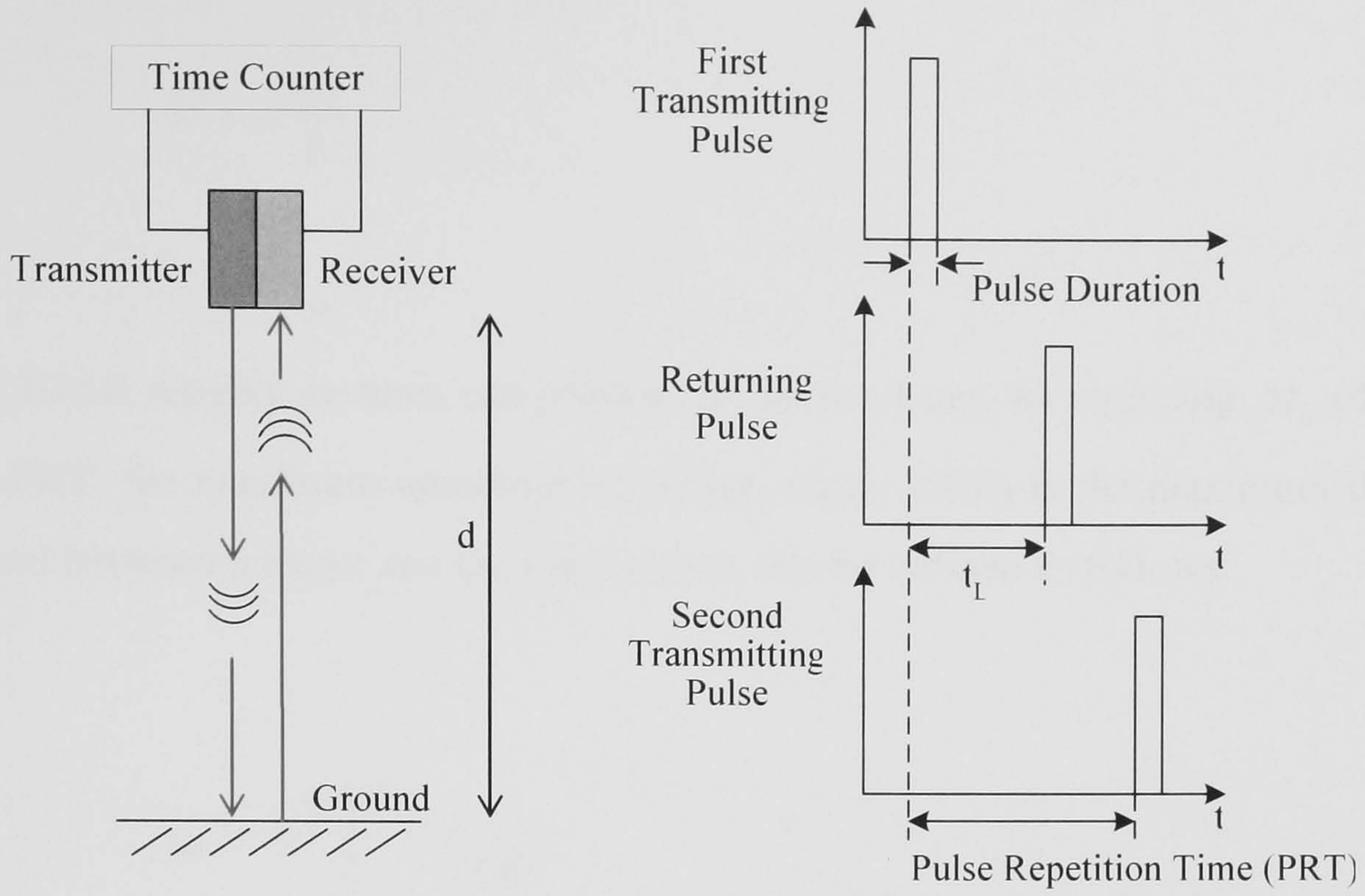


Figure 2-10. Schematic diagram of a laser raging system

Once a pulse has been transmitted by the transmitter, the range to a target is determined by first obtaining the total time elapsed between the time the pulse was transmitted and when it was received. This amount is then multiplied by the speed of light and that quantity is divided by 2. The factor of one-half is required since we are concerned with the range to a respective target and not the total (or roundtrip) distance traveled by the pulse. In equation form, this can be written as

$$r = c\frac{|t_t - t_r|}{2} = c\frac{t_L}{2} \tag{2-6}$$

where, r is a range from the transmitter to the target; c is the speed of light (~$3x10^{-5}$ km/s); t_L is the time difference between transmitting time, tt, of a pulse, and receiving time, tr, of backscattered energy from the same pulse. From Eq. 2-6, the range resolution, Δr, which is the ability to distinguish between targets on the same azimuth but at different ranges, can be derived as being proportional to the time measurement resolution, Δt_L. This can be described by

$$\Delta r = c\frac{\Delta t_L}{2} \tag{2-7}$$

Some LIDAR ranging systems can provide Δr up to ~1 cm. As replacing Δt_L of Eq. 2-7 with PRT, the maximum unambiguous range, rmax, which is the maximum distance measured between a target and the laser sensor, can be defined as follows:

$$r_{\max} = c\frac{PRT}{2} = \frac{c}{2PRF} \tag{2-8}$$

For a LIDAR system with 25 kHz PRF, the maximum unambiguous range is 6 km, which is beyond the flying height of typical airborne LIDAR systems (<1 km). However, the flying height must be balanced by considering other factors such as laser power, beam divergence, and sensor attitude errors, which degrade the overall system accuracy as the flying height increases.

In Eq. 2-7, the time interval, t_L, is determined by a time counter, which registers specific points on the transmitting and receiving pulses as t_t and t_r respectively. These epoch times are detected when the signal voltage of the pulse has reached to a pre-determined threshold value, typically 90% of its maximum amplitude on a rising side of the pulse, which is called as the leading edge. In reality, the leading edge is difficult to determine since the pulse received is not a rectangle as depicted in Figure 2-10. Since the transmitted has encountered many obstacles such as water vapours, windows, trees, and roof materials following its roundtrip optical path, the reflected amplitude is attenuated and the waveform generated has been distorted. Thus, the returning pulse cannot be maintained at the same level as the transmitted pulse. For this reason, the reflected echoes are amplified before they are sent to the time counter. However, the measured time, tr, will be delayed if the reflected echoes have widely varying shapes and causes variations in the leading edge detection. This case often occurs over uneven surfaces such as spruce stand, corn field, etc.

Another important parameter of the ranging system is the ranging accuracy. Ranging accuracy depends on the rise-time of received pulse, which is inversely proportional to signal-to-noise ratio. The pulse rise-time is a time duration required for a received signal to change from a specified low value to a specified high value. Typically, these values are 10% and 90% of the maximum signal amplitude. The signal-to-noise ratio depends on many factors such as the received optical power, input bandwidth, background radiation, and performance of signal detectors. The ranging accuracy, $\sigma_{R_{pulse}}$, of an received pulse can be defined by

$$\sigma_{R_{pulse}} \approx \frac{c}{2} t_{rise} \frac{\sqrt{B_{pulse}}}{P_{R_{peak}}} \tag{2-9}$$

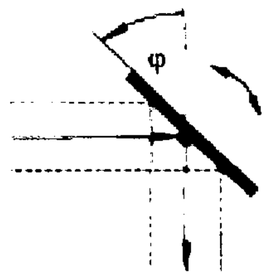
where t_{rise} is the pulse rise-time; $P_{R_{pulse}}$ is the received peak optical power; B_{pulse} is the input bandwidth (Wehr & Lohr, 1999). Eq. 2-9 implies that higher ranging accuracy will be achieved for a target emitting higher backscattering energy, while on the same target, a LIDAR system with higher laser power provides higher ranging accuracy.

Scanning system

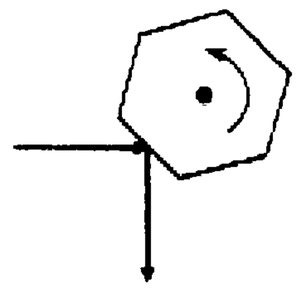
Ranging measurements obtained by a laser sensor, only looking down to the ground in the nadir direction would result in a single line on the ground along to the flight direction. To cover wider areas with higher point density, the laser beam is deflected across to the flight direction. The deflection of the laser beam is facilitated by a range of opto-electrical mechanisms, which results in special scanning patterns in two dimensions. The scan pattern on the ground depends not only on the laser scan pattern but also the flying direction and speed and the terrain topography. Typical scanning mechanisms employed by airborne LIDAR systems are shown in Figure 2-11. A general discussion of the laser scanning systems can be found in (Baltsavias, 1999; Wehr & Lohr, 1999)

Oscillating mirrors allow the scanning system to operate in uni-direction or bi-direction. Bi-directional mirror with two axes usually produces a zig-zag line, while a uni-

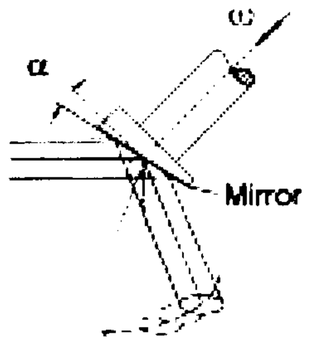
directional mirror generates parallel lines or curves. The oscillating mirror has to change the scan direction at each end of the scan, where its speed should be zero. Thus, the oscillating mirror normally requires an acceleration and deceleration to move mirrors, which affects the positional accuracy of the system.



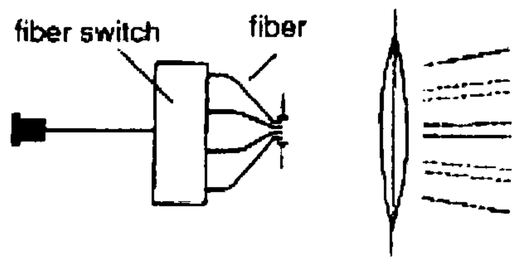
(a) oscillating mirror



(b) rotating mirror



(c) Palmer scanner



(d) fiber scanner

Figure 2-11. LIDAR scanning systems

The rotating mirror (or polygon) moves only in one direction with a constant angular speed, which produces the parallel scanning lines. The disadvantage of the rotating mirror is timing when the data is collected since there is no turning position indicating the change of scanning direction. Also, it might be difficult to correct a scanning bias because the mirror rotates only in one direction. Palmer scanner has a nutating mirror which results in an elliptical scanning pattern on the ground. The elliptical scanning produces redundant raging measurements as the forward and backward range scanning are always partly overlapped. These redundant points on the same ground position can be favorably used for the system calibration. Fiber scanner has identical transmitting and receiving optical fiber arrays, which generates a parallel scanning line on the ground. The small aperture of the fibers can achieve high scanning speeds, but require very sophisticate calibration procedure.

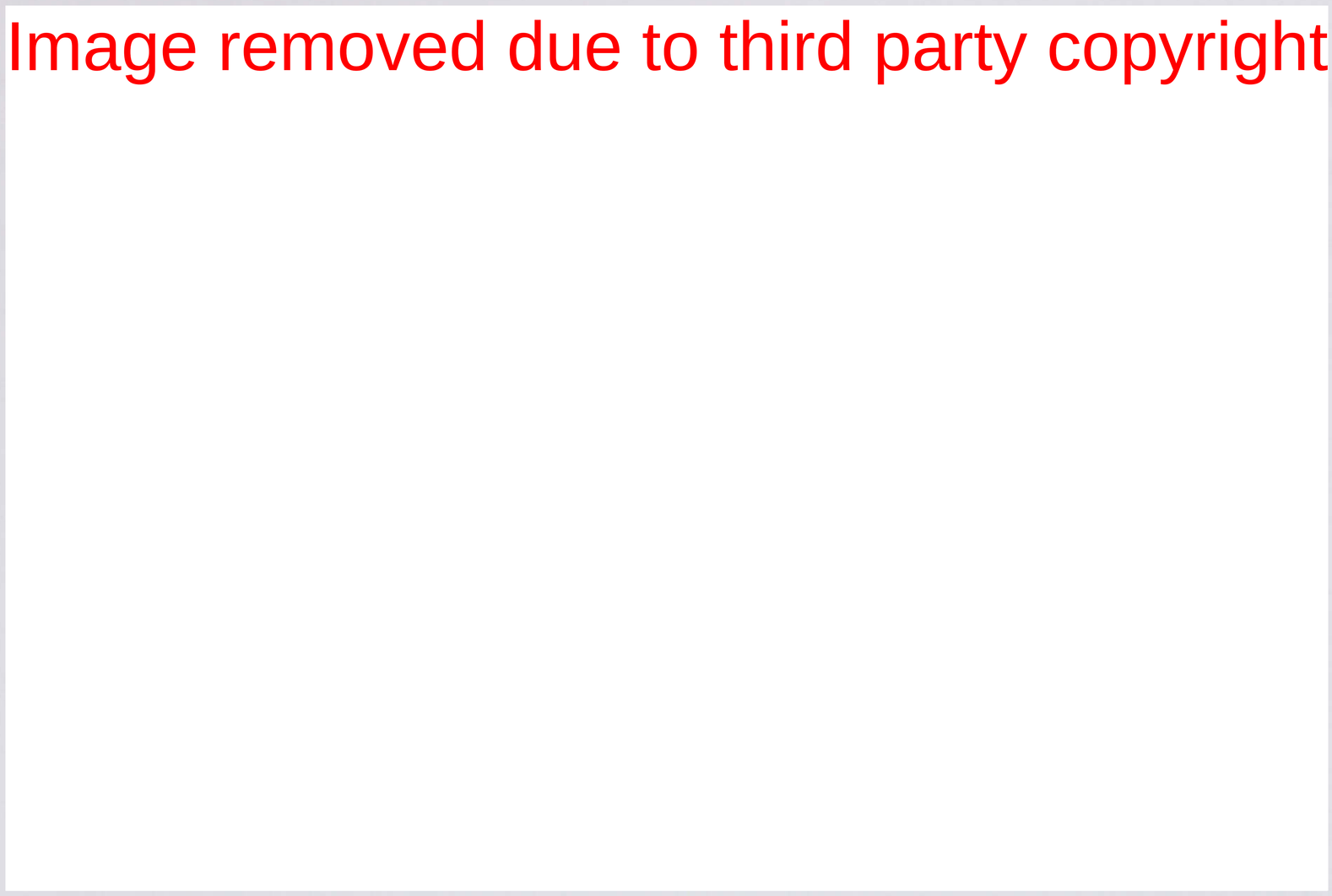


Figure 2-12. LIDAR ranging geometry (Dold, 2001)

By the aforementioned electro-optical scanners, the points along a line are usually scanned in equal angle steps, but their spacing on the ground is not constant. The narrow divergence of the laser pulse defines the instantaneous filed of view (IFOV). If the laser scanner is in the vertical position, the IFOV forms a circle on the ground. However, as the laser scan signal moves off the vertical, the IFOV will become elongated, forming an ellipse on the ground, along the scan direction. The maximum coverage of the scanning system can be achieved as the scan angle, θ, has reached to the system maximum, $\theta_{\max}$. At the flying height of 1 km, the scan angle normally operates in the range ±20°. Figure 2-12 illustrates the laser footprint, D, and the swath width, SW for a parallel scanning system. For a given flying height, h, the determination of SW and D mainly depends on the beam divergence, γ, and the scan angle, θ, as follows:

$$SW = 2h \tan\left(\frac{\theta}{2}\right) \tag{2-10}$$

$$D = \frac{h}{\cos^2 \theta} \gamma \tag{2-11}$$

The point density is of great concern for LIDAR surveying task. The point density achieved by a LIDAR system can be adjusted in several different ways by changing

PRF, flying height, flying speed, scan angle, and scan rate. PRF is usually non-selectable, but the scan angle and rate may or may not be adjustable for a particular instrument. Also, the flight-line side lap could be widened to increase the data point density. Considering an oscillating mirror system, the point spacing across and along flight direction, i.e., dx_{across} and dx_{along}, can be given by

$$dx_{across} = \frac{h}{\cos^2(\theta)} \frac{\dot{\theta}}{PRF} \tag{2-12}$$

$$dx_{along} = vt_{scan} \tag{2-13}$$

where $\dot{\theta}$ is the angular scanning rate, and t_{scan} is the period of one line scan and v is the flying speed (Wehr & Lohr, 1999). LIDAR systems on the market today provide a range of point density from 25 points per square meter to one point every 12 m (144 square meters).

Positioning and orientation system

The ranging and scanning system only measure relative distance and viewing direction from the sensor to the reflecting target on the ground. To obtain the 3D position of this point in a coordinate system such as WGS84, accurate position and attitude orientation of the sensor must be provided. To this end, all existing laser scanning systems are supported by the positioning and orientation system (POS), which consists of the differential global positioning system (DGPS) and the inertial measuring unit (IMU), often called as the inertial navigation system (INS). Figure 2-13 shows a local sensor coordinate system (x, y, z) centered at the laser pulse origin with three rotation angles and a local object coordinate system (X, Y, Z). The GPS provides the position of the laser sensor, $P_s(X_s, Y_s, Z_s)$, and the IMU provides the angular orientation of the sensor, (ω, ψ, κ), where ω is the angle around the flight direction (roll), ψ is the angle around the across-track direction (pitch), and κ is the angle around the vertical axis (yaw). With this information, the orientation and position of the laser sensor system can be

determined, thereby allow LIDAR system to geo-reference laser pulses reflected from the target, $P_t(X_t, Y_t, Z_t)$, to a local map projection.

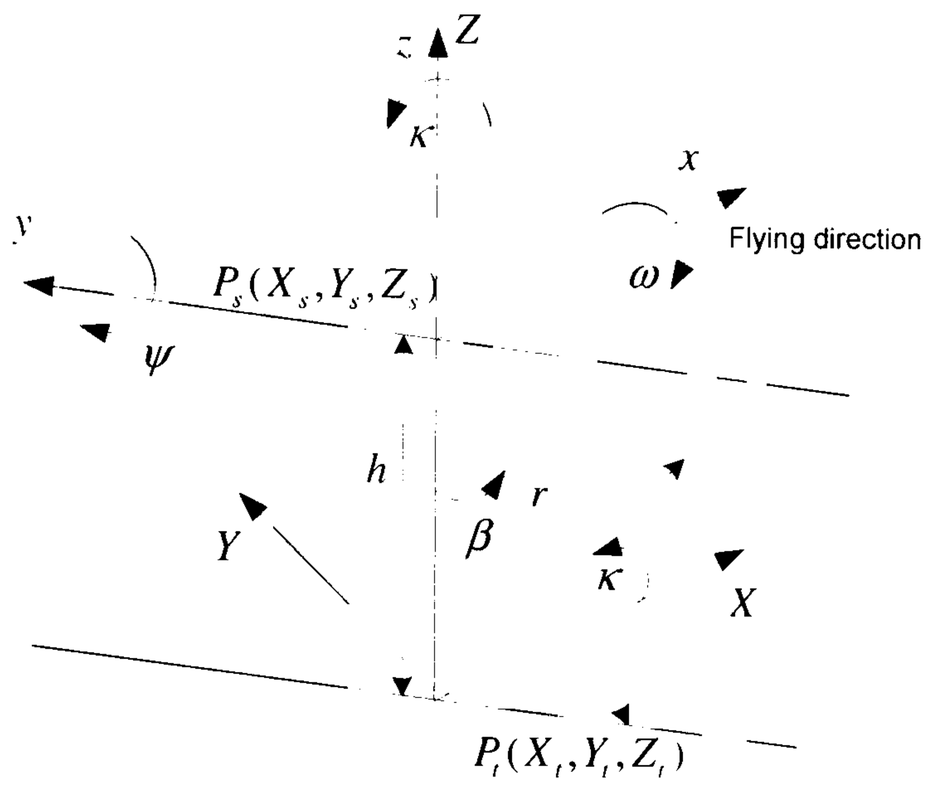


Figure 2-13. LIDAR sensor coordinate system with rotation angles

The GPS is a constellation of 24 satellites that orbit the Earth and transmit signals in the radio frequency. The GPS receiver mounted on an aircraft captures the GPS signals which provide timing and positional information to the LIDAR system. The laser pulses transmitted are GPS time tagged in order to later correlate them with DGPS solution. The GPS signal received allows one to calculate the distance (i.e., pseudorange) from the satellite to the receiver by very accurate timing system. With one satellite, a position of moving aircraft can be determined as a location on a sphere whose radius is equal to the pseudorange. As intersecting three spheres derived from three different satellites, a unique location can be determined for the receiver. In reality, a minimum of four satellites are required to estimate another unknown variable of clock bias. The GPS receiver is capable of measuring the L1/L2 carrier phase data at a rate of 1 Hz. The same type of GPS receiver is required within 30 km of the project area for ground control stations. The post-processing of GPS data between the receivers on-board the aircraft and the receivers on the ground control stations is known as On-The-Fly (OTF) Differential GPS. The DGPS solution allows for high-accuracy (<10 cm) 3D positioning of a moving platform. More than one ground control station is often used for providing redundancy and backup in case one of the GPS receivers fails.

The measurement of precise beam orientation is critical to correctly locate laser ranges in 3D space since the aircraft is tilting, pitching, accelerating or bumping during the flight. A very small change in attitude of the platform would result in a large change in horizontal distance. As the aircraft moves along its flight direction, the IMU, which consists of three gyros and three accelerometers, measures roll, pitch, and heading of the aircraft, establishing the angular orientation of the laser sensor about x, y, z axes in flight (see figure 2-13). This angular information is usually recorded at a high frequency of 50Hz. With this information, the location of the laser footprint on the ground can be determined. In contrast to GPS, the accuracy of IMU is time dependent, and less accurate orientation parameters are obtained over a long period of operation. Thus, the overall accuracy of IMU is less than the one of GPS.

2.3.4 LIDAR data processing

After a survey flight, two datasets comprising the POS data and the laser ranges are measured at instantaneous scanning angles. The first step is to download the GPS carrier phase data from the GPS receivers of the ground control station and the one mounted on the aircraft. These data are then processed by a GPS post-processing software package to compute the high-accuracy DGPS solution of the aircraft trajectory. The trajectory is then integrated with the IMU data by the Kalman filtering technique so that the GPS derived position is now transferred from the GPS antenna to the origin of laser sensor, which results in a complete set of exterior orientation data $(X_s, Y_s, Z_s, \omega, \psi, \kappa)$ for the sensor origin of figure 2-13. To this end, two mounting vectors should be accurately measured; i) the relative position of the IMU with respect to the GPS antenna, and ii) the relative position of the laser sensor with respect to the IMU. The laser ranging data and scanning angles are then merged to the exterior orientation in order to derive a geo-referenced X_t, Y_t, Z_t position of each reflected pulse measured by the sensor. This can be described by Kilian et al. (1996) as follows:

$$\begin{bmatrix} X_t \\ Y_t \\ Z_t \end{bmatrix} = \begin{bmatrix} X_s \\ Y_s \\ Z_s \end{bmatrix} + R(\omega, \psi, \kappa) \begin{bmatrix} 0 \\ 0 \\ r \end{bmatrix} \tag{2-14}$$

where r is the range distance measured from the sensor to the target. However, the above equation assumes that the LIDAR system has no systematic errors during its flying, which is not valid in practical operation. In fact, residual systematic biases in GPS, IMU and LIDAR systems would exist during the LIDAR surveying. These biases results in anomalies of the data acquired, resulting in a horizontal or vertical shift. Based on the research works conducted by Katzenbeisser (2003) and Schenk (2001), systematic errors causing the data anomalies can be summarized as follows:

- ***Ranging error*** - Time delays caused by the electronic components of the time counter leads to a ranging error. This delay might be a result of the unstable device temperature during the flight and the aging effect of the time counter. A more essential factor is the effect caused by a false adjustment of the oscillator frequency used in the time counter. Minor deviation from the operating frequency could cause reasonable errors of ranging measurement. Other error source can be found in the constant value of the speed of light. Since this value depends on the density of atmosphere, varying with pressure, humidity and temperature, a suitable value should be considered depending on the flying height.
- ***Sensor positioning error*** - The precision of the sensor position strongly depends on the accuracy of the DGPS solution. A DGPS error might be caused by the interruption of a GPS signal, the uneven distribution of rising and sinking GPS satellites, and the unstable conditions of troposphere and ionosphere. The DGPS errors occurred in the LIDAR surveying leads to the slight drift or even jumps causing difference between adjacent strips. If the survey exceeds one hour or extends for more than 30 km from the ground control station, the DGPS errors become obvious.
- ***Sensor attitude error*** - The IMU comprising gyro and accelerometers are not free of errors. Initialization of errors, a mechanical misalignment, and gyro drifts contributes to a systematic error of IMU. Even a minor error of IMU will lead to a drift in the attitude of the laser sensor, which can degrade the accuracy of the beam direction with respect to the IMU. In general, errors for roll and pitch in IMU will be from 0.004° to 0.02°, but the error for the heading is about twice the roll error (Katzenbeisser, 2003).

- ***Scanning error*** - A scanning error would be caused in case an instantaneous angular position of the scanner is not correctly recorded. This error might be caused by a mechanical misalignment of the scanner, resulting in angle offsets between the scanning plane and the flight direction or its vertical axis. Also, a false encoding of angular scanning positions by a zero-shift within an electronic device must be considered. The scanning error leads to the inaccurate measurement of the beam deflection vector with respect to the scanner, which causes a horizontal or vertical shift.
- ***Mounting error*** - The distance and orientation of the laser sensor with respect to the IMU might be changed during the flight due to some bending force on the carrying plate of the IMU and the laser sensor. The mounting errors, especially in remaining orientation errors are difficult to be removed and affect the position of the footprint proportional to the flying height. The same problem must be considered between the GPS antenna and the IMU. Since the center of the GPS antenna is difficult to establish a physical representation in a local IMU coordinate system, larger mounting error would be expected than the one of the laser sensor.

A calibration procedure determines the various systematic errors mentioned above and provides means to correct them. To this end, a calibration data must be collected by the LIDAR system as it scans over certain reference sites before and after the LIDAR surveying in different direction. The reference sites are usually flat terrain such as large soccer field, stadiums, buildings and building corners. Based on the calibration data, systematic errors causing data anomalies are analyzed. Since so many different error sources are correlated each other and various types of systems generate different errors behaviours, there is no standard calibration procedure. As a result, the raw data cannot be calibrated by the end users with convenient desktop software, but by the data vendor's established in-house process. Due to the lack of an adequate error model of LIDAR systems, systematic errors often remain in the calibrated data, and their effect become obvious when adjacent LIDAR strips are integrated with each other. An adjustment technique to correct remaining systematic errors varies from simple vertical or horizontal translations to remove systematic bias (Burman, 2000; Mass, 2000) to complex block adjustments that involve tie points among strips (Vosselman & Mass,

2001), and modelling systematic errors (Crombaghs et al., 2000; Filin, 2003; Huising & Pereira, 1998).

2.3.5 Applications

The direct products of LIDAR technology are high quality digital surface models (DSMs), which are formed by the point clouds reflected from the Earth's surface or the tops of man-made or natural objects. With very little or no post-processing, the DSM can provide 3D landscape models for fly- or walk-through visualization, landscape design, and computer game production.

A most definite application of LIDAR survey is to generate a digital terrain model (DTM) that contains the ground surface only. For this purpose, a specific filtering algorithm is required to remove non-terrain objects (e.g., buildings, cars, trees, etc) from LIDAR DSM, which will be discussed in §2.4. Airborne LIDAR systems are capable of rapid and accurate collection of elevation data with very high density. These features make LIDAR surveying a cost-effective method to generate DTMs. The DTM derived from LIDAR data can be used for a range of applications such as ortho-rectification of imagery, contour map generation, hydrological modelling, flood risk assessment and urban planning. Approximately 60% of the commercial activity conducted today is focused on the generation of DTMs (Flood, 2001). The Netherlands is the first country to have a complete country-wide DTM acquired by LIDAR, namely AHN ("Actual" Height model of the Netherlands) (Elberink et al., 2003; Huising & Pereira, 1998; Wouters & Bollweg, 1998). The AHN has an average point density of 1 point per $16m^2$ or better and a height precision of about 15 cm (standard deviation).

The potential of LIDAR systems for mapping forest stands has been intensively evaluated in the past few years. Accurate information both on the terrain and topography beneath the tree canopy can be achieved by the foliage penetrating capability of LIDAR pulses. This property allows the LIDAR data to be classified as trees, while the ground return serves as generating DTMs. By differentiating trees from the terrain surface, forest structure parameters (e.g., tree height, crown diameter, crown base diameter, tree density, biomass estimation, and etc) can be estimated. The forest applications using LIDAR data have been successfully demonstrated by Anderson et al. (2002), Morsdorf et al. (2003), Pyysalo & Hyyppae (2002), and Wack et al. (2003).

A promising application of LIDAR technology is the 3D reconstruction of buildings for 3D city modelling. Since LIDAR systems provide dense elevation data with high accuracy, detection and extraction of 3D buildings with sharp discontinuities are easier than in DSMs obtained by aerial photogrammetry, or faster than manual collection. Weidner & Forstner (1995) introduced a prismatic building modelling from a LIDAR DSM. The DTM is generated from a DSM by a morphological filter, from which the DSM is normalized with a threshold value. An isolated building region is approximated to the prismatic building model using MDL approximation. Brenner & Haala (1998) derive parameters for 3D polyhedral building models by least-square adjustment, which minimizes the distance between a LIDAR DSM and corresponding model. Based on the analysis of invariant moments, Mass & Vosselman (1999) presented closed solutions to generate parametric building models from a LIDAR DSM with a point density of 4-5 points per square meter. The technique showed even roof details such as dormers are achievable. Vosselman (1999) presented a data-driven approach to reconstruct 3D building shape with a high complexity.

LIDAR data has been used for corridor mapping application, which provides rapid collection of long, linear objects such as roads, railway tracks, pipelines, waterways, coastal zone or power lines. Since LIDAR systems have a narrower swath in comparison to optical sensors, they are more cost-effective in capturing information needed for the above applications.

2.4 LIDAR filtering techniques in literature

2.4.1 Introduction

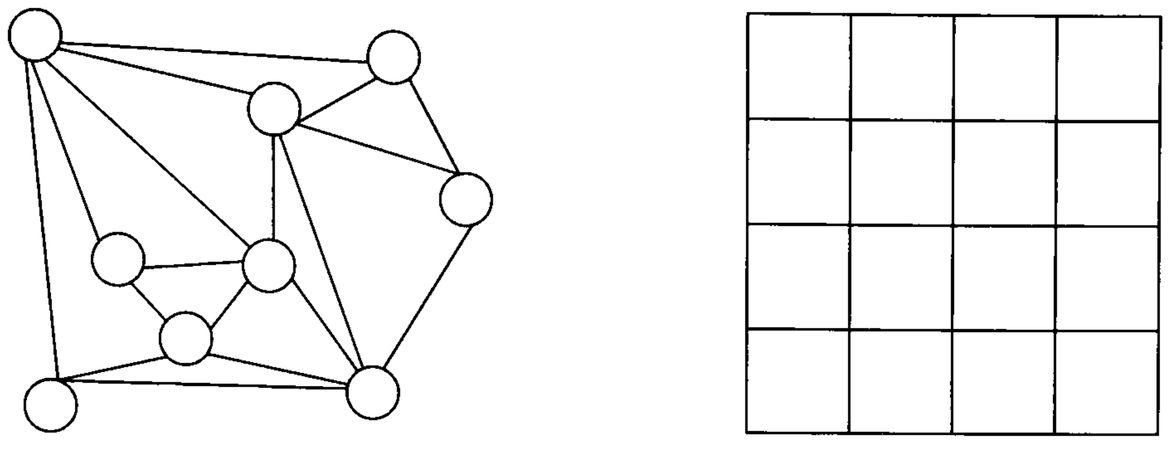
As has been discussed in §2.2, the generation of digital terrain models (DTMs) is the most definite application of LIDAR surveying, due to its high-density, high-accuracy and foliage penetrating capability. The elevation data acquired by LIDAR systems provides a digital surface model (DSM), which represents all the heights of objects on or above the terrain. If non-terrain objects such as vegetation and buildings are excluded from the DSM, the DTM can be generated. The LIDAR filter is a computer-based technique to automatically detect and remove those non-terrain objects, thereby generating the DTM. This section describes LIDAR filtering techniques which have been reported in literature. The first part describes different data structures used in

LIDAR filtering algorithms. In the remaining parts, various principles and mechanisms used for developing the LIDAR filter are discussed based on literature.

2.4.2 Data structure

After a series of complex system calibration was made as described in §2.2, a cloud of elevation data acquired by LIDAR system could be formed in one of two different data structures for further filtering purpose; a Triangulated Irregular Network (TIN) or a regular grid. The characteristics to store raw information and query topology are very different between the two data structures. Accordingly, LIDAR filters specialise in dealing with either irregularly spaced points in a TIN or regularly spaced points in a grid; it is rare for a LIDAR filter to process both of data structures.

A TIN is a data structure based on an irregular array of points which form a sheet of non-overlapping contiguous triangle facets. In TIN structure, the height values of LIDAR measurement are stored at individual sample point locations along with their planimetric coordinates, and the sample points are connected to each other by lines so that a well conditioned triangular network can be formed. TIN-based data structures have several advantages over grid data structure. Most important is that input data locations and height values are preserved in a TIN structure, and no transformation takes place. As a result, LIDAR filtering results achieved from TIN can avoid data loss. In addition, breaklines can be easily combined with TIN to augment significant discontinuities in landscape (McIntosh & Krupnik, 2002). TINs are also a native structure used for 3D graphics rendering, and thus, can be used for interactive 3D visualization of filtering results. Due to these reasons mentioned above, some LIDAR filters (Axelsson, 2000; Vosselman, 2000) are developed based on the TIN data structure. However, the creation and maintenance of TIN topology is computationally expensive since degrees of topology implemented in a TIN are high. For this reason, the computing speed of TIN-based LIDAR filters is slower than the grid-based filters.



(a) TIN (b) Grid

Figure 2-14. Illustration of TIN and grid data

A grid is a 2D rectangular array of cells, each of which stores an elevation value. The elevation values associated with the grid cells are obtained by sampling multiple points located in the vicinity of each grid cell. The disadvantages of the grid are that precise planimetric locations of LIDAR points are lost in the resampling of the grid, and also, sampling errors in height can be introduced in each cell when the height value is interpolated from neighbouring points. Thus, those sampling errors affect LIDAR filtering performance. The magnitude of the interpolation error introduced depends on gross and systematic errors in neighbouring points, the number of points involved in sampling process, the size of grid cell, and the interpolation algorithms used. More detail discussion about the interpolation errors of LIDAR measurements can be found in Lloyd & Atkinson (2002) and Smith et al. (2003). Although the LIDAR filtering result obtained from grid data is generally considered to be less accurate than the result from the TIN data, some LIDAR filters have been developed based on the grid data structure. This is caused by the fact that the grid data is favoured in many commercial and business environments, mainly as a result of the speed of processing and accessibility of software which requires regularly spaced data (Smith et al., 2003).

2.4.3 Filtering principles

LIDAR filters aim to distinguish points located on buildings and on the vegetation (off-terrain points) from those that are expected to be on the terrain (on-terrain points). This section describes key principles used for discriminating between on-terrain and off-terrain points, thereby generating the terrain surface.

Filtering criteria

LIDAR filtering is a classification process to distinguish on-terrain points from off-terrain points from a cloud of LIDAR measurements. At the end of filtering, the DTM

can be constructed by either removing off-terrain points, or detecting on-terrain points. To this end, suitable criteria, constraining geometric properties of on-terrain and off-terrain point (or maximizing their contrast), must be employed in filtering process.

Most of algorithms rely on a vertical height discontinuity as the filtering criterion, based upon the assumption that off-terrain points are generally located over the terrain with a certain height discontinuity. There are several variations of how to measure the height difference. As the simplest way, the height discontinuity was measured in the first order of neighbouring points; measuring the distance between a point and its neighbouring points in a grid (Baltsavias et al., 1995; Berthod et al., 1995; Haala, 1994). A geodesic distance is measured from the horizontal plane, which can be generated by a point with the lowest height within a block (Kilian et al., 1996; Lohmann et al., 2000; Masaharu & Ohtsubo, 2002; Petzold et al., 1999). Other height discontinuity can be measured as the distance from interpolated surfaces (Brovelli et al., 2002; Elmqvist, 2001; Kraus & Pfeifer, 1998), or as the distance from a TIN facet (Axelsson, 2000; Baillard & Maitre, 1999). Based upon the height difference measurements described above, a binary classification of on-terrain and off-terrain points is determined by a height discontinuity criterion given.

A slope measured between two consecutive points can be used as the discontinuity criterion, assuming that a point with abrupt change in slope is located on non-ground surface (Roggero, 2001; Sithole, 2002; Vosselman, 2000). The advantage of using the slope as criterion is that the terrain surface is better represented in slope, than in height, and a topographic analysis over certain area can be involved in the slope-based filtering algorithm. In addition, the slope measurement implicitly considers the point density used (the slope is changed depending on the distance between two points considered), which is useful to process irregularly spaced data.

Some LIDAR filters detect on-terrain points by removing off-terrain points (Haugerud & Harding, 2001; Lee & Schenk, 2002; Sithole, 2003a; Wack & Wimmer, 2002). Since non-terrain surfaces of vegetated and man-made objects generally show more roughness than the natural terrain, a local surface curvature is used as criterion to classify off-terrain points. Heugerud and Harding (2001) developed an algorithm to filter tree points in forest areas by comparing local curvatures of point measurements. Ground

measurements were selected by removing tree vertices iteratively from a TIN constructed from LIDAR measurements.

Deterministic vs. Adaptive filtering

The criteria described above imply the definition of natural terrain surfaces adopted by the filters. The critical problem in using the criteria is that the generic shape of natural terrain is hard to define. In fact, slopes of natural terrain contained in the landscape can be formed with various magnitudes, and parts of the terrain have similar geometric characteristics with the objects on the landscape; break-lines in the terrain such as cliff edge are geometrically similar to building edges. Moreover, off-terrain objects, such as buildings, cars, bushes, and trees, have free forms in real world. For example, planar surface roofs show similar smoothness to terrain surfaces.

The "deterministic" filter simplifies underlying terrain shape as being homogenous over the entire data set, i.e., dominated by a single terrain slope. Based on this assumption, a criterion, separating on-terrain and off-terrain points, is given by heuristic knowledge over a study area, which does not changed during the process. This filter can be applied to almost flat urban areas. However, the performance of the deterministic filter is significantly degraded even when it is applied to certain areas with small amount of surface variations, which frequently occur in the real world. The deterministic filters reported in literature can be found in Berthod et al. (1995), Weidner & Fostner (1995), and Vosselman (1999).

In the contrast to the deterministic filter, the "adaptive" filters drive a filtering criterion directly from the data used. Therefore, several different criteria can be locally employed for classifying LIDAR points depending on changes of underlying terrain slopes. This ability makes the filter adaptive to different landforms, which is useful property when the filter applies to urban or forest areas where heterogeneous terrain slopes usually co-exist. However, the filtering performance relies on how to reliably analyze underlying terrain characteristics, thereby driving the local filtering criteria. This adaptive filtering strategy has been adopted in Axelsson (2000) and Kraus & Pfeifer (1998).

Local vs. global process

In the local filtering process, filtering criteria, removing off-terrain objects or detecting on-terrain features, always operate on a local neighbourhood. The neighbouring points play an important role to enable the filter to analyse the characteristics of an underlying terrain surface. Thus, part of terrain information must be driven from the neighbouring points considered. However, the size of neighborhood is difficult to determine in the filtering process since the size of off-terrain objects and the change rate of terrain slope are generally unpredictable.

In contrast to the local filtering process, global processing derives terrain surface characteristics from the entire data sets. It enables the filter to avoid the locality problem caused by the use of pre-determined neighbouring size. A global energy function (Baillard & Maitre, 1999) or a Fourier series (Belli et al., 2001) is driven from the entire data sets in order to approximate the terrain surface. However, the global approach is limited to reconstruct almost flat terrain, or require *a priori* topographic knowledge, since it is difficult to formulate a global optimization function for complex terrain with no constraints.

Single vs. iterative process

Some filter algorithms classify points in a single pass while others recursively classify points in multiple passes. The single step algorithm needs less computation than the iterative process. However, the iterative process makes the filter gather more information about the neighbourhood of a point, by changing either the neighbouring size or filtering criteria in each iteration. Thus, a much more reliable classification can be obtained. Most of adaptive filters employ the iterative process.

2.4.4 Filtering mechanism

A filtering mechanism to facilitate the LIDAR point classification can be designed depending on the strategy used to integrate the various filtering principles mentioned in the previous section. This section categorizes a range of LIDAR filters reported in literature depending on their filtering mechanisms and discusses their pros and cons.

Thresholding techniques

Thresholding techniques assume underlying terrain as being either a flat or a plane surface. Based on that, a single height threshold only is required to differentiate terrain

surface from others. Due to its simplicity, the techniques are rapid to reconstruct terrain surface, but limited to fairly flat terrain surfaces. Baltsavias et al. (1995) suggested using a histogram-based segmentation technique. DSM heights are grouped into consecutive bins (i.e., height ranges) of a certain size, called MHB (Multiple Height Bins). The MHB serves as segmenting the DSM into relatively few regions, giving closed polygons. The terrain regions are obtained by thresholding the MHB. The selection of the height threshold from MHB was crucial for achieving a high quality DTM, but it was not provided in their work.

An alternative solution focused on relative positions between regions has been proposed in Gabet et al. (1997). The surface is segmented into homogeneous regions using a classical region growing algorithm based on local height discontinuities. Then the segmented regions are sorted into two groups according to their relative height discrepancies. The terrain segment is obtained if its mean height is lower than certain threshold from neighbouring regions. However, the classification process is very local and could fail over buildings with complex structure, or urban areas with high-degree of terrain discontinuity. Similar approaches can be found in Berthod et al. (1995) and Cord et al. (1999).

Morphological filtering

Mathematical morphology employs two fundamental operations (i.e., "dilation" and "erosion"). Dilation and erosion are commonly used for enlarging (dilate) and reducing (erode) the size of features in binary images. The concept of dilation and erosion can be extended to find the minimum and maximum value by a specified window kernel in grayscale images. With an elevation data measured by a LIDAR system, the dilation output is the maximum elevation in the neighborhood of a point, whereas the erosion replaces the elevation value of a point with the minimum in its neighbouring elevation values. The combination of the erosion and dilation generates an "opening" operation, which is employed to filter LIDAR data. As an elevation of a point is compared to its neighborhood within a given window size, the opening operation eliminates all local maxima in height, while the dilation restores the shape of terrain surface.

Weidner & Forstner (1995) successfully demonstrated that the DTM generation can be achieved by the morphological opening operation. However, the selection of a filtering

window size is critical for the success of this method. If a small window is used, most of ground points will be preserved, but building objects of sizes larger than a given window will not be removed. On the other hand, the filter tends to over-remove the ground points with a large window size. Ideally, the window size of the morphological filter should be small enough to preserve all ground details and large enough to remove non-terrain objects (Zhang et al., 2002). Unfortunately, there is no single optimal value for the morphological filter.

This problem can be overcome by introducing a series of opening operations with different window sizes. Kilian et al. (1996) suggested an iterative opening filter starting from the smallest window size, which becomes larger in later iteration. A weight related to the window size is assigned to each point classified as on-terrain point. The larger the window size of an operation, the higher the weight of a point. Finally, the terrain surface is generated by using all the on-terrain points classified with assigned weights. Instead of using weight function, Zhang et al. (2002) introduced an elevation difference threshold based on elevation variations between the opening iterations. An opening morphological filter with the smallest window size generates an opened surface. An elevation difference map is computed by eliminating the opened surface from the original elevation data. The ground points for which elevation differences are lower than an elevation threshold are obtained. In a second iteration, the filtering window size increases and the elevation threshold decreases. The same process used in the first iteration is applied to the previous filtered data. As the filtering iteration proceeds, more terrain details will be preserved, but larger buildings will be eliminated. Although they suggested a function to determine the elevation thresholds related to the window size, the function requires *a prior* knowledge of an expected terrain slopes over study area.

Block-minimum algorithms

A block-minimum algorithm assumes that on-terrain points should be the lowest points in a neighborhood. Kilian et al. (1996) proposed a block-minimum function to remove off-terrain points. After an opening morphological filter is applied to the data, a horizontal plane is driven by a point with the lowest height within a given window size. Then on-terrain points are detected if their heights fall within a pre-defined height buffer from the horizontal plane. All the on-terrain points are detected by moving the filtering window over the entire dataset. Similar block-minimum filter, which has been

used in the company TopScan, was introduced by Petzold et al. (1999). An initial terrain surface is first generated by obtaining the lowest points in a moving window. Then, off-terrain points are removed if their heights exceed a given threshold from the initial terrain surface generated. Masaharu & Ohtsubo (2002) suggested a statistical block-minimum filter. A study area is divided into a number of regular interval patches, which size is normally 20 (m^2) or 50 (m^2), and select the lowest points in each patch as on-terrain points. For each on-terrain point, a point buffer with a pre-defined radius (normally, 50 or 100 metre) is generated to collect its neighbouring on-terrain points. Then, the mean and standard deviation is computed, and a point is removed if it is lower or higher than one sigma level from the mean height. This process iteratively continues until the statistics of all the points becomes small enough for the process to terminate. Finally, DTM is generated by interpolating remained on-terrain points.

The block-minimum filter implemented in the grid data is computationally rapid. However, block-minimum algorithms have two fundamental weaknesses. Like the morphological filter, the selection of the optimum block size seriously affects the overall filtering performance, which is usually hard to determine. On the other hand, block-minimum algorithms contain implicitly assumption that the ground is horizontal. The technique is limited to flat or low slope areas.

Slope-based filter

Vosselman (2000) proposed a slope-based filter that detects on-terrain points by comparing slopes between a LIDAR point and its neighbours. The slope-based filter defines the maximum allowed height difference between two points as a function of distance, namely erosion function. When this function is known, a virtual terrain surface is generated according to the distance between a point and any other point within a given circle. This point is classified as on-terrain point if all the member points considered are located above the virtual terrain surface.

The implicit assumption of applying the slope-based filter is that there is a distinct difference between the slope of terrain and that of off-terrain objects such as trees and buildings. The critical step in slope-based filtering is to determine an optimum erosion function, i.e., the maximum value of slope allowed. Determining the slope threshold in various landforms is somewhat subjective. Vosselman (2000) demonstrated that good

results could be obtained by using slope threshold from training data sets. However, the training data sets have to include all types of on-terrain points in a study area to achieve good results, which is not always practical. Also, the filter suffers difficulties to optimize the kernel size of erosion filter, i.e., the size of neighbouring points considered. If the kernel size is smaller than half the size of a building, the building is not completely removed by the slope based filter. Vosselman & Mass (2001) suggested these artifacts remained could be eliminated by a heuristic post-processing. Zhang et al. (2002) reported that satisfactory result can be achieved by applying the slope-based filter to flat urban areas such as Miami, Florida. However, the classification errors were large when this method was applied to vegetated mountain areas with a large slope variation.

Sithole (2001) improved the performance of the slope-based filter in steep slopes. In this modified slope-based filter, a slope map is first prepared by selecting the lowest points from each cell of subdivided grid, and the slope-based filter can refer to the slope map as to obtain a local sloping trend. However, the slope map includes off-terrain features as well as on-terrain points, which degrades the overall reliability of the slope map. Moreover, determining the optimum cell size is another issue. Sithole (2001) reported that the improved slope-based filter can preserve steep slopes, but it still faces problems in removing large buildings and areas with low height vegetation.

Roggero (2001) suggested another idea to make Vosselman's slope-based filter adaptive to various terrain slopes. The local slope is estimated by a linear regression method. For a point, each neighbouring point within a local operator given is compared to the lowest point in terms of distance and height difference. Then, the height differences measured are weighted according to the relative distance from the lowest point, which is used as observation in a linear regression. As a result of the process, the maximum slope allowed for the point is determined, which feeds back to the slope-based filter. Roggero (2001) demonstrated a robust result in an urban area. However, a number of parameters for weighting function and linear regression must be determined in advance.

Iterative TIN densification

Axelsson (2000, 2001) suggested an adaptive TIN model to detect on-terrain points in urban area. A sparse TIN is created as an initial terrain model by selecting the lowest

points within a user-defined grid of a size greater than the largest non-ground features. Then, one point above each TIN facet is added as on-terrain point if its parameters are below a data-driven threshold. The iteration continues until no points can be added to the terrain model. Several parameters for a point are measured as three angles to corner points of the base triangle (or surface normal of the base triangle), and the orthogonal distance from the TIN facet. In every iteration, a data-driven threshold is determined based on a median value estimated from the histogram of point parameters collected.

The main strength of this algorithm lies in the ability of dealing with surface discontinuity, which is particularly useful characteristic in urban area. However, the use of a median value as the classification criterion requires relatively large number of on-terrain points must be located in each TIN, which may be difficult to control as the TIN becomes dense with smaller size, especially in urban area. In addition, the filter assumes on-terrain points in a study area are located on similar types of terrain in terms of terrain slope. Thus, different thresholds have to be given for various land cover types. A commercial software implemented based on this method can be found in the TerraScan software (TerraScan, 1999).

Iterative linear prediction

Kraus & Pfeifer (1998, 2001) proposed an interpolation method based on a linear prediction iteratively to remove off-terrain points and generate DTMs. An initial terrain model is generated by selecting the lowest points from subdivided patches. From selected points, an average surface is created, and then, the residuals from the average surface to the measured points are computed. Each point measurement is given a weight according to its distance by a non-symmetric weight function; a point above the average surface is given small weight, while a large weight is assigned to a point below the surface. The average surface is now interpolated by a linear interpolation (Kraus & Pfeifer, 1998), considering the weights assigned. Only the points, which are located below or above the interpolated surface within a predefined threshold, are selected for iterating the aforementioned process. This process is repeated until the generated terrain surface does not change significantly between iterations, or the maximum number of iteration is reached. This interpolation method has been extended to adopt a hierarchical strategy, where only points detected from coarser level of terrain surface are considered for generating the next (i.e., finer) level of terrain surface. This algorithm has been

implemented in the SCOP software, which can be found in Pfeifer et al. (2001) and Briese et al. (2002).

The algorithm requires a "good" mixture of on-terrain and off-terrain points so that a weighting function is reliably parameterized. If the filter is applied to a large bare field with very a small number of off-terrain points or even with no off-terrain point, parts of bare field located on hilly side or hill tops may be reduced to flat area, since those elevations are considered as noise by the patch-wise weighting function. Therefore, the iterative linear interpolation is not guaranteed to converge when being applied to urban areas, where the mixture of on-terrain and off-terrain points is unpredictable within each patch. In addition, the technique is basically based on an interpolation method, which makes the underlying terrain model favourable smooth surface. However, this property is hard to satisfy in urban areas where significant discontinuity of natural terrain exists.

Energy minimization

The role of an energy function in minimization-based algorithm is two fold: (1) as the quantitative measure of the global quality of the solution, and (2) as a guide to the search for minimal solution (Li, 2001). As the quantitative measure of costs, with which an algorithm reaches to the solution, an energy function defines the global solution as its minimum cost. In regard to the LIDAR filtering problem, a global solution is determined when the filter obtains all the on-terrain points with the minimum cost.

Elmqvist (2001) suggested an active shape model to obtain the terrain surface. This algorithm is based on a general technique of matching a deformable shape to a gridded DSM. An energy function deforming the terrain shape is defined as being linearly combined with three different energies, i.e., internal, external, and potential energy. The three energies imply an expected smoothness of the reconstructed terrain surface, deforming rate and "favorable" terrain shape respectively. An initial deformable plane is generated with a horizontal plane driven from the lowest point of small sub-region and the energy function is computed. Then, points with the minimum cost to deform the initial plane are collected as on-terrain points, and the initial deformable plane is updated. This process continues until the total energy monotonically decreases.

Baillard and Maitre (1999) described a terrain surface as the Markovian model, where all the points are defined as random variables, each of which is conditioned by its neighbouring points. A global potential function is defined, which consists of the data attachment potential and the contextual potential function. When a label is assigned to an arbitrary point as either on-terrain or off-terrain label, the global potential energy is calculated for the point, considering its labelling cost. The global potential energy becomes minimized if a proper label is assigned to the point. The process will be terminated if no local labelling change decreases the global potential energy.

Belli et al. (2001) modelled natural terrain with a Fourier series, which captures the main characteristics of the terrain surface. The terrain surface is modelled with a decomposition of orthogonal Fourier harmonic functions. The Fourier coefficients are then rigorously estimated using Turkey's M-estimator iteratively. Although they first suggested means a parametric model approximating natural terrain, the initial guess of the Fourier dimension (i.e., the number of terrain slopes expected over study area) must be known as prior information.

Clustering

The clustering or segmentation technique aims to subdivide the entire LIDAR measurements into disjoint regions, each of which has homogeneous property separating from its surroundings. In this framework, more refined feature classes than a binary classification of terrain and non-terrain object can be obtained. There are various different ways to segment the entire LIDAR points into a set of homogeneous regions.

Filin (2002) defined feature classes to segment natural terrain into four categories, i.e., high vegetation (forest), low vegetation (rough surface), smooth surface, and planar surface. In this method, clustering is performed based on an attribute vector consisting of the point position, the tangent plane to the point, and the height difference of a point and its neighbours. The three attribute vectors measured for all points generate a feature space, from which pre-defined surface classes are identified. With detected feature classes, all the points are again clustered in object space by proximity measures. Then, each point cluster is tested by pre-defined thresholds, i.e., the minimum size of the point cluster and the maximum standard deviation in the clustered feature attributes. Those point clusters exceeding the thresholds are dismissed or split if needed. This process

continues until no further points can be added. After iterative clustering process, those clusters that share similar attributes are merged and tested for final surface model. The method demonstrated a good clustering, even using a low ground spacing of 2.5m.

The advantages of this technique is that the classification can be achieved with a surface modelling, which can be benefit for further detail object recognition such as tree and roof modelling. Unlike conventional data segmentation, the algorithm does not require a seed point selection, and moving window operation for feature grouping. However, on-terrain points are sparsely distributed around trees in forest areas. In this case, the suggested method might suffer difficulties to form smooth surfaces and fail to reconstruct terrain surface. In addition, the technique did not suggest how terrain surface can be extracted from the clustering map. A merging process to recognize an object such as natural terrain is another issue. In fact, this is an inherited problem of general clustering (or segmentation) algorithms since a natural object is hard to define by *a prior* assumption.

2.5 Building reconstruction in literature

2.5.1 Introduction

Automatic detection and modelling of building objects has been a major topic of research in computer vision and photogrammetry communities for many years. A wide range of approaches has been published in literature (CVIU, 1998; Gruen et al., 1995; Gruen et al., 1997).

In general, automatic building modelling can be subdivided into three independent tasks; detection, extraction and reconstruction. These tasks can be defined as follows:

- ***building detection***: recognize the existence of buildings of interest and localize them into regions.
- ***building extraction***: delineate physical building boundaries in 2D or 3D, which implicitly includes the building detection problem.
- ***building reconstruction***: find building roof structures and delineate a complete description of building roof shapes in 3D object space as well as building boundaries.

The tasks mentioned above form a logical hierarchy for the acquisition of 3D building models. Figure 2-15 illustrates this. The result of building detection can be used as an input for supporting the building extraction or the building reconstruction process. Similarly, 3D reconstruction of a building roof can start from its boundary delineated by the building extraction process. In some cases, however, these tasks can be achieved independently of one another. For example, it is possible to perform building boundary fitting on regions with no indication of building location. Also, it is not strictly required that building extraction must be performed before the building reconstruction. Building boundaries can be obtained during or after building roof shape is reconstructed. However, most building reconstruction systems follow the logical hierarchy of the three tasks mentioned earlier. The reason for this is that the complexity of building reconstruction problem can be reduced greatly as each task filters off distracting features from the observed data in its own purpose. Thus, each process can focus on its own task within much simplified data domain.

The detection and delineation of building objects is of interest for many applications. It can greatly improve the automation of 2D building map generation which, as a result, can be used in various applications including urban planning, telecommunication, risk assessment, etc. Moreover, the performance of building extraction seriously affects the result of building reconstruction, which, in the worst case, provides wrong semantic or geometric description of buildings. The process of building extraction varies a lot. It depends on supporting models and data sources used. This section reviews some of the recent developments reported in the field of building extraction algorithms. It should be noted that the reconstruction of 3D building models is beyond the scope of this review. Successful examples of building reconstruction can be found in Fischer et al. (1998), Haala & Brenner (1998), Haala & Hahn (1995), Jamet et al. (1995), Noronha & Nevatia (2001), Weidner & Forstner (1995).

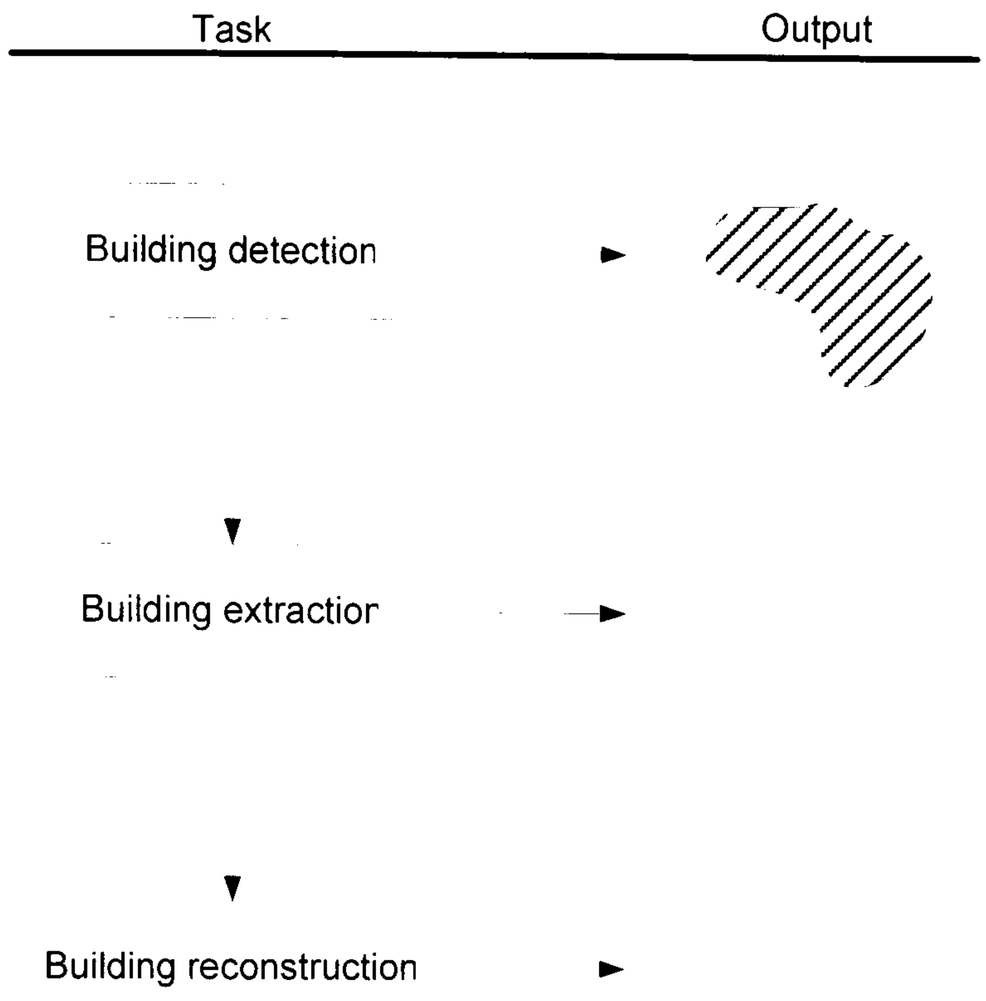


Figure 2-15. Illustration of a hierarchical building reconstruction tasks

2.5.2 Building object representation

Generic knowledge about the shapes of buildings is represented using a range of building models. There are two different models for the description of the building objects, i.e., parametric, and generic model.

The parametric building model uses pre-specified model instances including rectangle, "L"-shape, "I"-shape, "T"-shape, gable roof, and hip roof (see Figure 2-16). The number of modelling cues such as corner points, straight lines, and surface patches to form a model instance is pre-determined, and their parametric relations, which are usually angles and relative distance between the modelling cues, are given from the model database. However, the geometry and topological relations of the modelling cues are unknown. Once the modelling cues are extracted by feature extraction algorithms from data sources used, these are matched with the pre-determined parametric model in images or a DSM to estimate the correct geometry of the model and thus, reconstruct the building instance, referred as building hypothesis. In fact, the parametric model provides a set of cue grouping rule. To model more complex buildings, the building models used are merged through a set operation such as intersection, union, and subtraction.

The parametric modelling approach provides specific object interpretation for internal structure and building classification, which is useful for some applications. In addition, the use of pre-specified models can infer modelling cues, which are often missed by feature extraction algorithms. However, since the number of building models used is pre-fixed, the parametric modelling approach is of limited application. Indeed, building shapes in an urban environment usually shows high irregularity, which is unpredictable. These irregular building shapes cannot be represented by the parametric models. Since the geometry of the parametric model is unknown, too many modelling cues can be matched with the model. As a result, the number of building hypotheses generated is very large, and the selection of a correct building representation is always a bottleneck of this approach. For this reason, most parametric building models are used for the reconstruction of residential houses, whose shape variances can be represented by using one or two model instances.

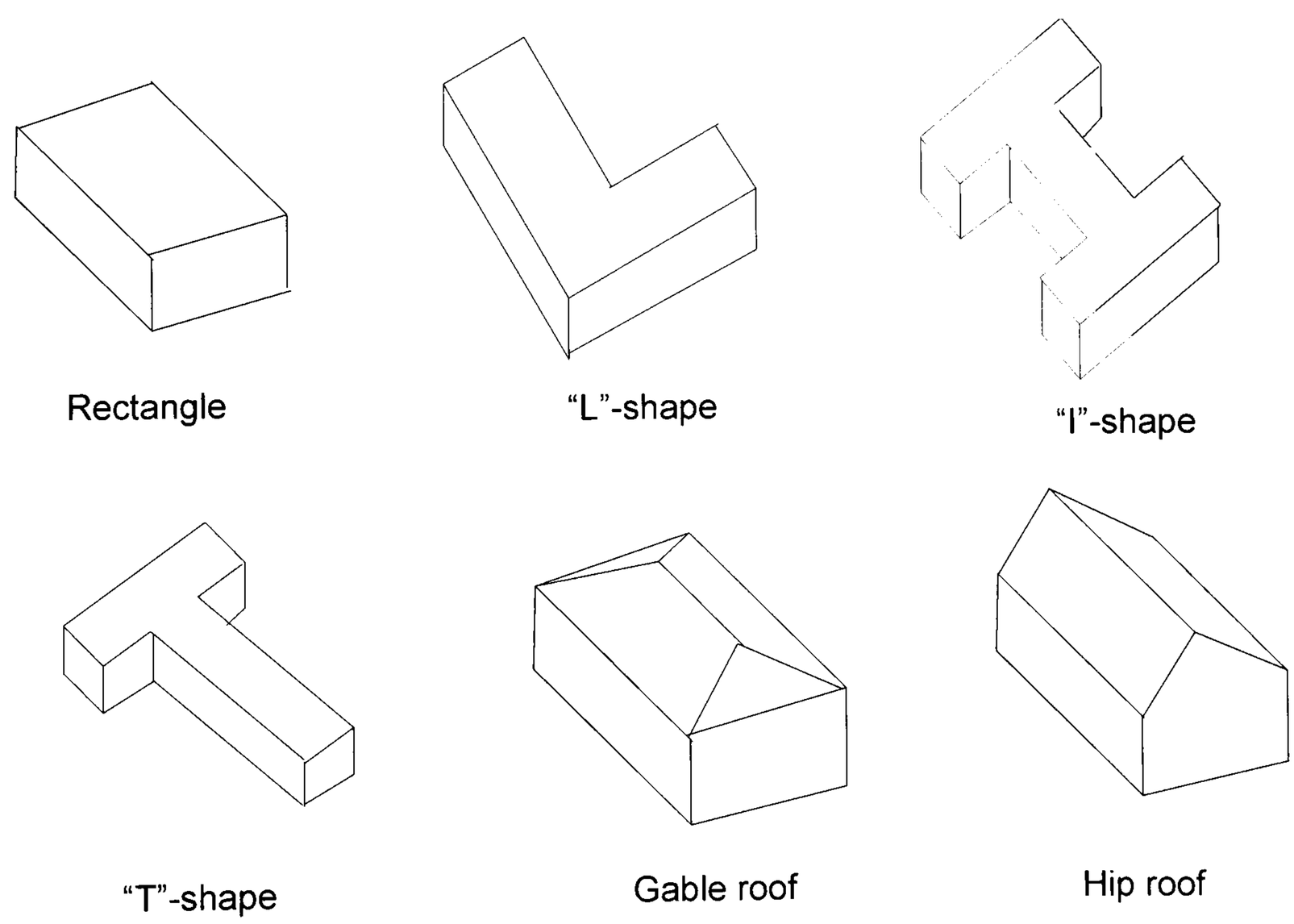


Figure 2-16. Parametric building models

In contrast to the parametric building model, the generic building model does not rely on pre-specified model instances. Instead, the model specifies an internal structure of building object only using a set of geometric parameters which is not pre-fixed in number. A building shape is reconstructed by geometric grouping of modelling cues extracted by feature extraction algorithms. This approach is suitable for representing

much irregular building shape with minimal shape constraints. However, the major problem of generic models is that buildings can be reconstructed completely only if feature extraction algorithms cannot detect all the modelling cues required. This problem often occurs in an urban environment since urban scenes normally contain rich object context with many occluding parts, which disturbs a "clear" object interpretation. In this case, the missing cues cannot be recovered by the generic building model since generic models do not use *a priori* knowledge of building shapes, and thus the completeness of building representation is hard to achieve. For this reason, most generic modelling approaches uses multiple images to increase the modelling cue density, or a DSM with very high vertical accuracy to extract planar patches which covers all parts of the building rooftops. The generic building model has two sub-classes; prismatic model and polyhedral model. The prismatic model represents a building shape with a set of parametric edges and an associated height for the entire building. This model does not provide a roof structure. The polyhedral building model forms a roof structure in 3D by collecting polygonal surface patches. The internal relations of roof structure are obtained by intersecting the polygonal surfaces extracted. Figure 2-17 shows some examples of the generic building models.

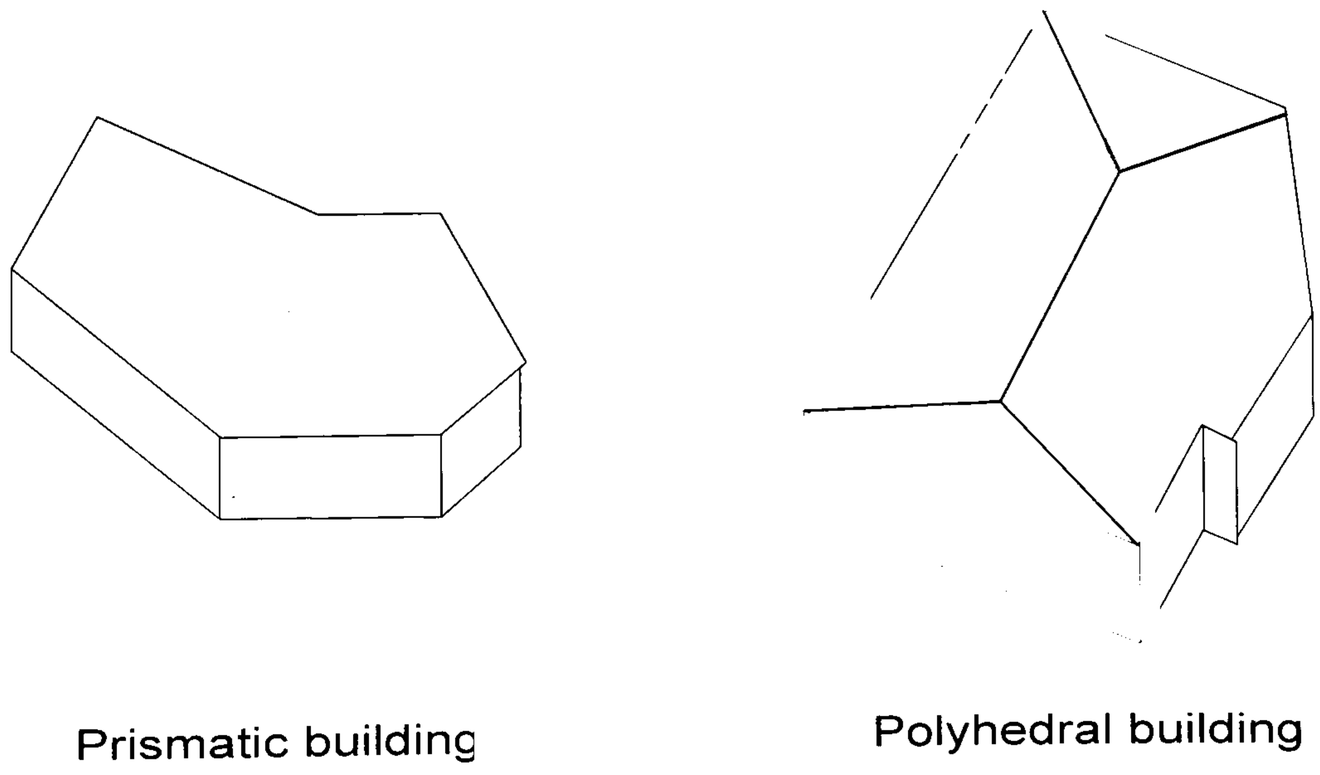


Figure 2-17. Generic building models

2.5.3 Using monocular image

Most of the early approaches toward building extraction focused on extracting building boundaries from a monocular image. Since a building extraction technique with

monocular images requires a minimal use of data source, it has been an active research topic (Shufelt, 1999). Restriction to a single imagery, however, often makes it more difficult since modelling the extracted cues cannot be used for deriving 3D analysis. It is difficult to decide which cues belong to real building structures, without using 3D information. As a result, modelling the generated cues usually presents a considerable amount of errors. In this regard, a strong assumption of the building shape, surface reflectance or textures is required to reduce the size of cue space, and additional knowledge such as shadow and wall evidence is required to verify "real" building cues.

Mohan & Nevatia (1989) developed a building extraction system using monocular image based on a specialized cue grouping technique known as "perceptual grouping". It is assumed that building outlines are structured with a great deal of geometric regularity, by which the extracted features can be grouped by the geometric relations such as proximity, collinearity and symmetry. The system starts to detect straight lines, which are further grouped to form more complex parametric cues. With the lines are extracted, parallels, "U"-shapes, and rectangles are subsequently identified until all the rectangles are generated. After this, a rectangle with the strongest supporting evidence and minimum conflicts is finally selected as a building model. The supporting evidences include the edge coverage of lines, the number of corner points, the number of crossing lines, the amount of overlap between parallel lines, the amount of texture inside the rectangle. The selection process of the rectangular building model was implemented in a "Constraint Satisfaction Network". Unfortunately, the network was not efficient to handle with a large number of modelling cues.

Lin et al. (1994) improved the method by filtering off distracting model hypotheses at an earlier stage. A local selection is applied to the hypothesized rectangles to choose those that are more likely to correspond to buildings. The local criteria evaluate local supporting evidence such as lines, corners, and their spatial relations. A score for each hypothesis is computed by using all local selection criteria and only those exceeding a threshold are retained for global selection. Global selection criteria evaluate relationships such as containment and overlap among the remaining hypotheses which allow some of them to be eliminated. With the knowledge of the sun direction, shadow and walls are used for finally verifying the selected hypotheses.

Jaynes et al. (1994) proposed different cue grouping and verification process for extracting planar polygonal rooftops in a monocular image. Orthogonal corners and lines are extracted from images, where the corners are used as the primary cues for generating building hypotheses and lines are used as supporting evidences for verifying them. Those features are grouped in a "feature relation graph". Orthogonal features serve as nodes in the graph and binary relations between corners are represented by a line between the corresponding nodes. A closed cycle in the graph corresponds to a plausible building hypothesis. The technique also considered a partially opened cycle as the building hypothesis. In this case, the missing features around the opened area are searched in top-down manner. If evidence is sufficiently found in the image, the opened cycle turns into the closed one by generating "virtual" corners and lines. A final building boundary is delineated as the maximally weighted and closed cycle is extracted in the graph.

Kim & Muller (1999) made a similar graph-based approach to detect buildings in monoscopic images. In this approach, straight lines are detected using a conventional edge detector and the extended up to predetermined length following their line direction. Then, a "line-relation-graph" is constructed by assigning types and values to connections between extended lines. Finally, the closed sub-graphs are found within this graph and are used for delineating most outlines of building boundaries. Even though lack of reasoning process and model details is often criticized, the method can delineate more complex building shapes by considering only the mathematical and geometric relations between the extracted lines.

The techniques mentioned above assume that the images are taken from a "nadir" view where only a building roof is visible. Shufelt & McKeown (1993) proposed a building extraction method to handle monocular images with general viewpoint. They used vanishing point information to derive the perspective geometry of 3D structures. Given the position of the camera and the sun, shadow geometry and illumination analysis can be utilized to verify the presence of buildings. These constraints play a role in reducing the search space of modelling cues for generating building hypotheses. Noronha & Nevatia (1997) used the geometric and projective constraints to make a hypothesis for the presence of building roofs from low-level features and to verify this by using available 3D cues. Their building model is restricted to rectilinear buildings with flat

roofs. At first, linear edges are detected from the image. Then, parallelogram hypotheses are formed, which are consistent with the projective constraints given by the viewing geometry. Promising hypotheses are selected based on some 2D and local 3D evidence. The selected hypotheses are verified by searching for 3D cues using wall and shadow evidence. The verified hypotheses are examined for mutual containment and overlap. Finally a non-conflicting set is selected, which provides 3D building models.

The building extraction techniques using a single image are mainly based on low-level features such as lines and corners. However, it is very difficult for feature extraction algorithms to extract all the significant parts of buildings. The methods limit the representation of building to rectangles or their simple combination. Thus, it may fail to perform with urban scenes which normally contain buildings that are not rectangular in shape. Moreover, the acquisition of the verification evidence required is subjective to the scene complexity, oblique viewing angle and the image resolution used.

2.5.4 Using multiple images

When multiple images are available, two major advantages can be achieved for building reconstruction. On the one hand, 3D information can be associated with modelling cues, which supports a 3D analysis of building structures. On the other hand, multiple-views can provide redundant information and occluding building parts can be recovered in the other images. For these reasons, many building reconstruction systems use stereo or multiple images. With given 3D information, 3D corners, 3D lines and 3D planes are mainly used as primary cues for building reconstruction.

Fischer et al. (1998) developed a hierarchical feature grouping method, in which symbolic 2D image description is converted into 3D description of building objects. A Feature Adjacency Graph (FAG) is used for representing the 2D image description, where a 2D corner is associated with its connected lines and regions. Those 2D corners are used for deriving 3D corners through multiple viewing geometries. A number of parameterized building parts are inferred from 3D corners, which serves as a "terminal" or "connector" to form a pre-specified building model. With the 3D corners reconstructed, building hypotheses are generated by merging and connecting the 3D building parts, and a final model is selected by verifying the model hypotheses in image space.

Bignone et al. (1996) developed the ARUBA system for 3D building reconstruction using multiple colour images. The building detection is achieved by manually marking a rectangular window enclosing the same building in all images. It extracts 2D lines from images and measures their radiometric, chromatic attributes and geometric relations. 3D location of the 2D lines extracted from a single image can be computed by obtaining their stereo-paired lines in the other images as well as by maximizing an "edginess" measure along the epipolar line. Then, the 3D lines are grouped to comprise 3D planes and 2D enclosures are associated with the 3D planes to form 3D patches. These 3D patches are scored according to the intersection constancy with their neighbouring 3D patches. Finally, 3D patches with "good" intersecting quality are retained for reconstructing a 3D building model.

Baillard & Zisserman (2000) introduced a polyhedral building reconstruction system. It starts with extracting 2D straight line segments in the image. Using a line matching algorithm they derive 3D edges from these images. The line matching algorithm (Schmid & Zisserman, 1997) not only exploits the geometry of multiple views but also the intensity information in the images to overcome the fragmentation of the edges. Starting from the 3D edges they use the intensity information on both sides of the 2D edges to find the optimal orientation of a half plane passing through the 3D edges. Extending and intersecting these half planes leads to complete the building roof shapes. The novelty of the system is to generate a planar patch hypothesis from a single 3D edge by maximizing the intensity similarity over multiple images.

Moons et al. (1998) used a polyhedral model for reconstruction of 3D roof structures. A constraint triangulation network is constructed from the 2D lines extracted from the image. A building region is isolated by manually selecting triangles that contain fairly homogeneous colours as a building roof. Then, 3D line segments are generated by matching line segments belonging to the same regions. These 3D lines are grouped to construct polygonal patches. After the adjacent polygons are integrated, a polyhedral roof model is constructed and its metric accuracy is improved by fitting the model to the image data. Scholze et al. (2001) proposed an improvement of Moons et al.'s work. They designed parameterized models of convex polygonal roof patches. More complex polygons can be modelled by merging those roof patches. 3D lines are extracted from multiple colour imagery, by which coplanar plane hypotheses are generated. Roof patch

hypotheses are instantiated from at least two coplanar 3D lines, and similar patches are iteratively merged. A final roof patch is selected through Bayesian estimation as the models are fitted to the images. The geometry of the roof model is adjusted to maximize the utility function which pre-encodes the shape preferences of roof patches.

With 3D information obtained from multiple images, these systems show that building models of 3D roof structures are achievable. The problem of the systems is that most of them require more than two images with very high resolution (less than 50cm), which are not often available. In addition, the systems suffer from difficulty in automation of building detection. Most systems resolved this problem by manual process, but it is very time-consuming.

2.5.5 Using multiple data sources

Since images usually contains very rich context, it is difficult to completely separate objects of interest from surrounding features. Under this circumstance, many distracting features that are not associated with building structure can be involved in the automatic building reconstruction process, which poses some difficulties in the building reconstruction using only aerial images. To reduce the complexity of the scene context, other data sources are often combined with images. These additional data sources are digital surface model (DSM) and 2D ground plans.

DSM can be generated by stereo matching from multiple images or measured by LIDAR systems. The fundamental idea of using DSM is to extract underlying terrain information, thereby inferring location of buildings. The basic assumption here is that buildings are located above the terrain with a certain height discontinuity and that some terrain segments exist between isolated buildings. As the terrain surface is extracted from the DSM, the location of buildings can be precisely computed. The performance of building detection entirely depends on the quality of DSM used. The result of building detection only provides rough building boundary information, and further building extraction process is required.

Haala (1994) used a DSM which is manually measured at one point each 5 x 5 square meter. Building regions are detected as the DSM is segmented by computing isolines in height. For each region, the attributes of mean height, roundness of shape and size are determined. Building regions are then detected by simply thresholding the region

attributes. 2D line extraction and 3D line matching process is made over each isolated building region. Parallel 3D lines are grouped within a pre-determined search space, and then lines perpendicular to these parallels are detected, which construct the parallelograms for building roof modelling.

Ameri & Fritsch (2000) combined a high resolution DSM with stereo aerial images for reconstructing 3D polyhedral building models. Building regions are detected based on a morphological top-hat information. Over each building region, geometric characteristics of surfaces are used to extract flat-pixel surface type. The extracted pixels serve as the seed regions for region growing algorithm, which partition the image surface into planar roof regions. A robust parameter estimation technique, combining random sampling consensus (RANSAC) algorithm with least median squares (LMS) method, is used for computing the planar parameters of each region. Then, the geometric relations of these 3D cues are described by a polygon adjacency relation (PAR) graph. Based on the PAR, compatible 3D polygons are merged, and coarse polyhedral building models are generated. Finally, the metric accuracy of the model is improved as the model is fitted to edge pixels of buildings by least-square method.

Cord et al. (2001) showed their ETIS system, which aims at extracting the polyhedral building models from aerial photographs with high resolution (less than 0.1m GSD). First, DSM is generated by the most up-to-date area-based stereo matching techniques using a geodesic adapted window. A classical region growing algorithm is applied to the DSM, which leads to the classification of on-ground or above-ground regions. By an analysis of surface normal distribution within the above-ground regions, individual buildings are separated from their surroundings. Over each isolated building region, a building roof with multi-slope planes is reconstructed by a Bayesian model selection process. This method does not restrict on the distribution of the different planes. However, the maximum number of planes comprising building roofs is limited.

In some countries, ground plans providing a very accurate building detection are available. By relatively simple interpretation of the ground plans, the building detection problem can be resolved, and useful information about the shapes of buildings can be inferred. In addition, the ground plans provide focusing areas, where the process of

feature extraction or 3D matching is restricted. As a result, the hypothesis and verification space are greatly reduced.

Pasko & Gruber (1996) proposed a fusion technique combining stereo aerial images with 2D ground plans. 2D straight lines are extracted from one of the stereo images. The ground plans of the buildings are matched with 2D lines based on affine matching process. The matching process requires the knowledge of the approximate height of the buildings and the ground. The result of the fusion process is enhanced by the involvement of a second image, which generates 3D roof lines within the ground plan of the buildings. Starting from the extracted outlines of the building, ridge lines and other edges of the roof are detected. The final roof model is generated by manually editing detected roof lines.

Brenner & Haala (1998) reported on an automatic building reconstruction system, which relies on LIDAR DSM and existing ground plans. First, the building ground plan is subdivided into a number of rectangles. Then a 3D parametric building model including flat, hip, gable and desk roof model is selected for each 2D rectangle based on analysis of the DSM. Then the parameter of the selected model is estimated over the DSM. Finally, all 3D models are assembled in order to describe the entire building structure. Relying on the rectangular partioning result may limit the practical use of the technique since general architectural partitioning is more complicated. Brenner (2000) improved the method, in which the ground plans only provides building boundary polygons and the building reconstruction is mainly performed by the interpretation of LIDAR DSM. A LIDAR DSM is segmented by the RANSAC algorithm, which results in planar roof patches. Those planar patches comprising significant roof structures are determined by a rule-based filtering process. 3D roof edges are obtained by intersecting all retained planar patches. With the building ground plan, a full description of 3D building shape is achieved.

Vosselman & Suveg (2001) showed a different partition method for 3D building reconstruction. By extending the ground plan lines that intersect in concave corners, a building ground plan can be segmented into small rectangles. By using LIDAR DSM, the partitioning result of the building ground plans is improved. LIDAR DSM is segmented in planar roof patches by the Hough transformation, which served to detect

height jump edges over the roof. 3D lines detected refine the rectangular partitions of the building ground plans, and the final partitions can be obtained after merging similar roof patches. A parametric roof model is hypothesized and tested within partitioned rectangles, and a full description of the roof model is generated by collecting the verified parametric models. Suveg and Vosselman (2002) further extended their technique to interpret aerial stereo images instead of LIDAR DSM. Similar rectangular partitioning is applied to the building ground plans. The partitioned ground plans and parametric models are projected onto the images. A parametric model providing the maximum evidence between the projected model and images is selected as the final building model. Image gradients along the projected model contours and intensity similarity over two images are used as evidences for the model evaluation. The method , using stereo images, can be problematic if the 3D information extracted is inaccurate and the model projection onto the images includes errors.

Combining DSM with existing ground plans for a fully automatic reconstruction has shown good results, even in urban areas and on datasets with several thousand buildings (Brenner, 1999). The advantage of using the ground plans is to obtain very accurate result of building extraction, which provides information about building location and building boundary polygons. Thus, remaining building reconstruction becomes much simpler. However, as buildings become more complicated, the relation between roof structure and ground plan becomes more obscure and can possibly not be exploited anymore. Also, the existence of 2D ground plans must be presumed, which is valid in some countries.

2.6 Summary

This chapter has described the two primary data acquisition systems in great detail, of which datasets are used in this study. In addition to this, it has been attempted to sort out some advance and improvement made recently in LIDAR filtering and building reconstruction techniques. Based on this, the next chapters will demonstrate how the new algorithms have been developed and applied, and what could be improved for broader applications.

CHAPTER 3

AUTOMATED DIGITAL TERRAIN MODEL GENERATION FROM LIDAR DATA

3.1 Introduction

One of the research topics in this thesis is concerned with the derivation of a digital terrain model (DTM) directly from a digital surface model (DSM) acquired from the LIDAR sensor. In particular, the DTM generated from the LIDAR DSM will be used as a base layer for localizing buildings and generating building models. This chapter describes a filtering technique developed for automatically extracting terrain surfaces and removing non-terrain objects from LIDAR DSM. As discussed in §2.3, approximately 60% of LIDAR applications today are focused on the production of terrain surfaces (Flood, 2001). Over the last few years, various types of LIDAR filtering algorithms for generating DTMs have been reported (Axelsson, 1999; Elmqvist, 2001; Kraus & Pfeifer, 1998; Masaharu & Ohtsubo, 2002; Petzold et al., 1999; Roggero, 2001; Sithole, 2001; Vosselman, 2000; Wack & Wimmer, 2002; Zhang et al., 2002). There is, however, no unified filtering algorithm, which can be applied to all circumstances comprising different landforms with various slopes and irregularities (Schickler & Thorpe, 2001; Sithole, 2001; Vosselman & Mass, 2001). Hence, further manual classification which removes features unnecessary to reconstruct the DTM is still required, consuming an estimated 60 to 80% of processing time (Flood, 2001).

In this chapter, the major problem of the LIDAR filtering technique is described as a serious lack of "normalized" knowledge, which universally represents natural terrain and non-terrain objects. In fact, generic terrain surface continuously varies in slope with

different magnitudes, and non-terrain objects are formed in various sizes and surface structures. This ill-constrained object representation often requires the filter to rely on a large number of constraining parameters and limits its areas of application only to those satisfying the conditions favoured by the filter. This chapter introduces a new LIDAR filtering algorithm which is continuously adaptive to varying terrain slopes and the different sizes of non-terrain objects with a few parameters. The developed filter assumes that generic terrain surface is a mosaic of plane terrain surfaces. A plane terrain surface is modelled, and the DTM is reconstructed by recursively fragmenting a LIDAR DSM with a set of plane terrain models. In §3.2, the problems of the deterministic and adaptive filtering mechanisms are discussed. Based on that, a new approach is described. In §3.3, the key concepts of the developed filter are introduced, and the algorithm is presented in more detail in §3.4. In §3.5, each processing step of the filter is illustrated by using two different LIDAR data sets.

3.2 Statements of problems and new approach

3.2.1 Labelling problem

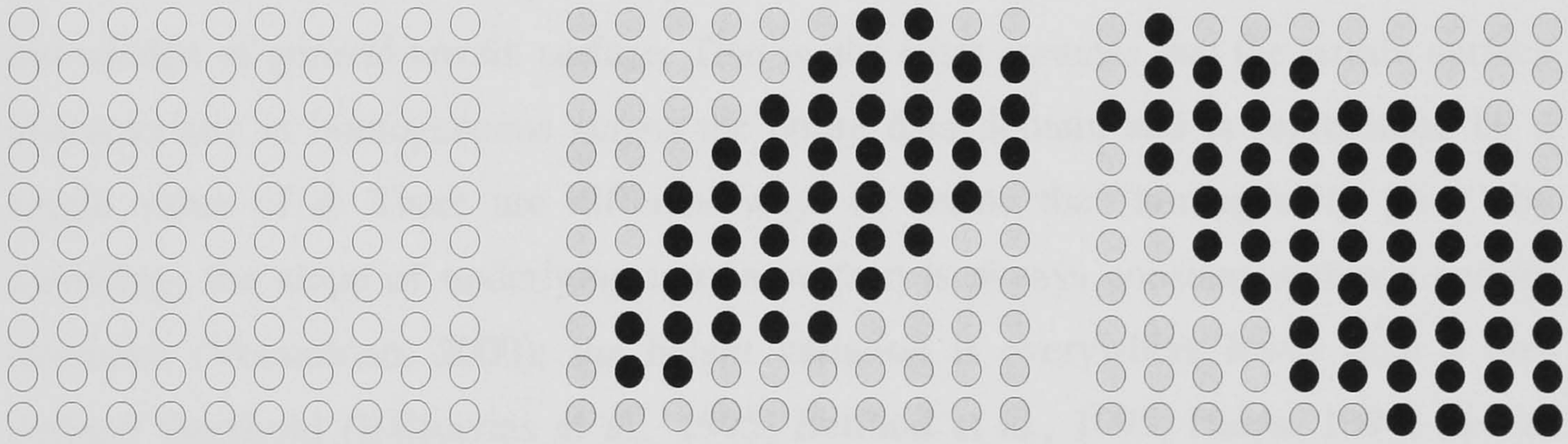
LIDAR filter is used for detecting on-terrain points and removing off-terrain points from a cloud of LIDAR measurements, thereby generating terrain surface model. This technique is often considered as a labelling process in which predefined semantic labels are assigned to data (Li, 2001). Suppose that we have a data domain, S, which comprises N LIDAR points (i.e., $S = \{s_i\}_{i=1}^{N}$). The LIDAR filter employs a labelling function, F, in order to determine the most adequate label for each point of s_i from a pre-specified label set, {on, off}. Here, {on} is a label assigned to an on-terrain point, which is located on terrain surface, while {off} is a label for an off-terrain point, which is located over non-terrain objects such as vegetation and buildings. By the labelling function, F, all the points will have their own labels, {on} or {off}, which leads to a label distribution of f as follows:

$$f = \{f_i\}_{i=1}^{N};\ f_i = \mathrm{F}(s_i);\ f_i \in \{on, off\} \tag{3-1}$$

where f_i is a label assigned to a LIDAR point, s_i, from a label set {on, off}. To activate the labelling function, F, in Eq. 3-1, it is required to select a criterion, δ, to differentiate on-terrain points from off-terrain ones. When the selected δ is associated with the labelling function, Eq. 3-1 can be re-described by

$$f = \{f_i\}_{i=1}^{N};\ f_i = \mathrm{F}(s_i|\delta);\ f_i \in \{on, off\} \tag{3-2}$$

As can be seen in Eq. 3-2, all the points are conditioned with the filtering criterion, δ, for determining an adequate label for each point. Thus, the quality of the filtering result depends on the selection of δ. Different filtering criteria will generate different label distributions as can be seen in figure 3-1.



(a) raw LIDAR points (b) label distributions

Figure 3-1. Illustration of a binary labelling function; the label distribution changes depending on the selection of δ; unfilled, gray, and black filled dot represent raw, on-terrain, and off-terrain label respectively.

The δ plays a role in the LIDAR filter to provide a geometric definition of generic off-terrain objects or natural terrain surface. The fundamental problem of LIDAR filtering is caused by the fact that neither labelling classes are well defined. Generic terrain surfaces are almost everywhere continuous and differentiable up to an unpredictable order, which cannot be restricted within a certain limited range. A cliff-edge may show similar geometric discontinuity to building boundaries, while a planar building roof show very smooth surface characteristics like a plain terrain surface. On the other hand, off-terrain objects include a range of different objects classes from small bushes to

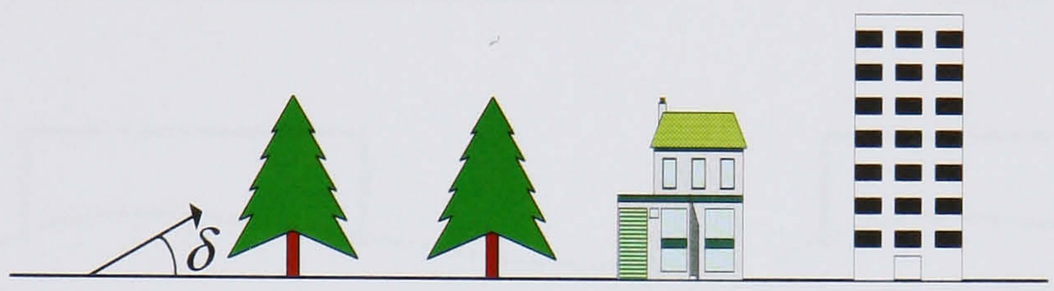
buildings, and they have different structures and sizes. Unfortunately, there is no universal definition of on-terrain or off-terrain label. For this reason, most LIDAR filters often produce some amount of labelling errors. The labelling errors of a LIDAR filter are quantified by the Type I and Type II errors. The Type I error is a terrain "omission" error, where a true on-terrain point is labelled as {off}, while the Type II error is a terrain "commission" error, where a true off-terrain point is labelled as {on}. Therefore, an optimum filtering result can be achieved as the filter minimizes both of Type I and Type II errors. In the following sections of §3.2.2 and §3.2.3, it will be discussed how both labelling errors are caused by two representative filtering mechanisms (i.e., deterministic and adaptive filters), and present a new approach to overcome their limitations in §3.2.4.

3.2.2 Deterministic filtering technique

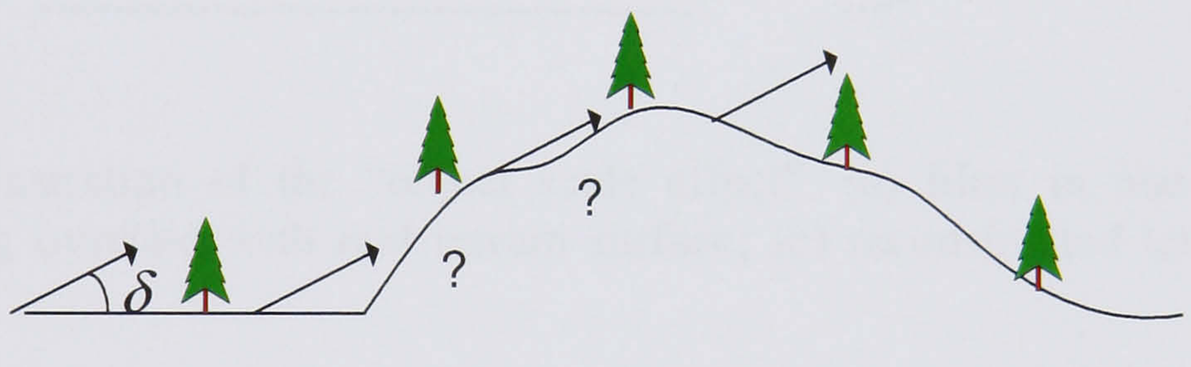
As described in §2.4, the deterministic filters are dominated by using a single criterion, δ, which does not change during the process. The use of the single criterion implies that the deterministic filters simply adopt the "homogeneity prior" for representing the knowledge of generic terrain surface. That is, the filter assumes that the terrain surface characteristic is homogeneous across the entire data domain and is represented by a single value of δ. There are different ways to define the "homogeneity prior" by assuming; the slope of underlying terrain surface is always constant within a certain tolerance (Vosselman, 2000); the height variation is everywhere lower than a pre-defined threshold (Baltsavias et al., 1995; Berthod et al., 1995; Haala, 1994; Lee & Schenk, 2002); the Laplacian property of terrain height is less than a specific threshold (Wack & Wimmer, 2002). With those terrain representations mentioned above, the filter classifies the LIDAR points by investigating whether or not the terrain homogeneity assumption is violated by δ.

There are, however, two fundamental drawbacks in the deterministic filter. First, it is difficult to constrain natural terrain by a single criterion and thus, the "homogeneity prior" assumption of the deterministic filter is significantly destroyed even with a small amount of slope variations. Figure 3-2 illustrates an example where a slope criterion, δ, is selected to differentiate on-terrain points from off-terrain ones over a flat terrain surface; a point with slope larger than δ is labelled as an off-terrain point, otherwise as an on-terrain point (see figure 3-2(a)). The δ is, however, only valid when it is applied

to the flat terrain surface. If the δ is applied to different terrain surfaces from the flat terrain, the δ leads to the generation of Type I error. As a result, real on-terrain points are mistakenly recognized as off-terrain points, since the "homogeneity prior" assumption of underlying terrain surface as being flat is not valid (see figure 3-2(b)). It suggests that the LIDAR filter must be adaptive to terrain surfaces varying with different magnitudes, namely "terrain variation effect".



(a) δ applied on a flat terrain surface



(b) δ applied on a terrain surface with the mixture of various terrain slopes

Figure 3-2. Illustration of "terrain variation effect"

Second, the deterministic filter also suffers from difficulties in defining off-terrain objects. The geometric properties of off-terrain features are not well constrained by using a single criterion. In general, off-terrain features such as power lines, trees, cows and buildings have "free-forms" with respect to the underlying terrain surface in terms of their spatial distribution, size and structure. Some parts of off-terrain features such as planar roofs may show compatible surface homogeneity with the one of terrain surface and thus, off-terrain points located over the planar roof surface are labelled as {on}, rather than {off}. Therefore, Type II errors are generated. Some filters such as the morphological filters (Weidner & Forstner, 1995) resolve this problem by using knowledge of the physical extent of the off-terrain objects, thereby determining the size of the filtering kernel. The problem of the morphological filter is the filter does not allow different sizes of off-terrain objects, but relies on a single kernel size, which is usually pre-determined based on *a prior* knowledge of the maximum building size. This

causes labelling errors if the kernel size is not adjusted to the real scale of underlying off-terrain objects. As can be seen in figure 3-3, the filter with large kernel size tends to over-remove terrain surface, while the filter with small kernel size under-removes underlying off-terrain objects. Therefore, the LIDAR filter must be adaptive to various scales of off-terrain objects, namely, "object scale effect".

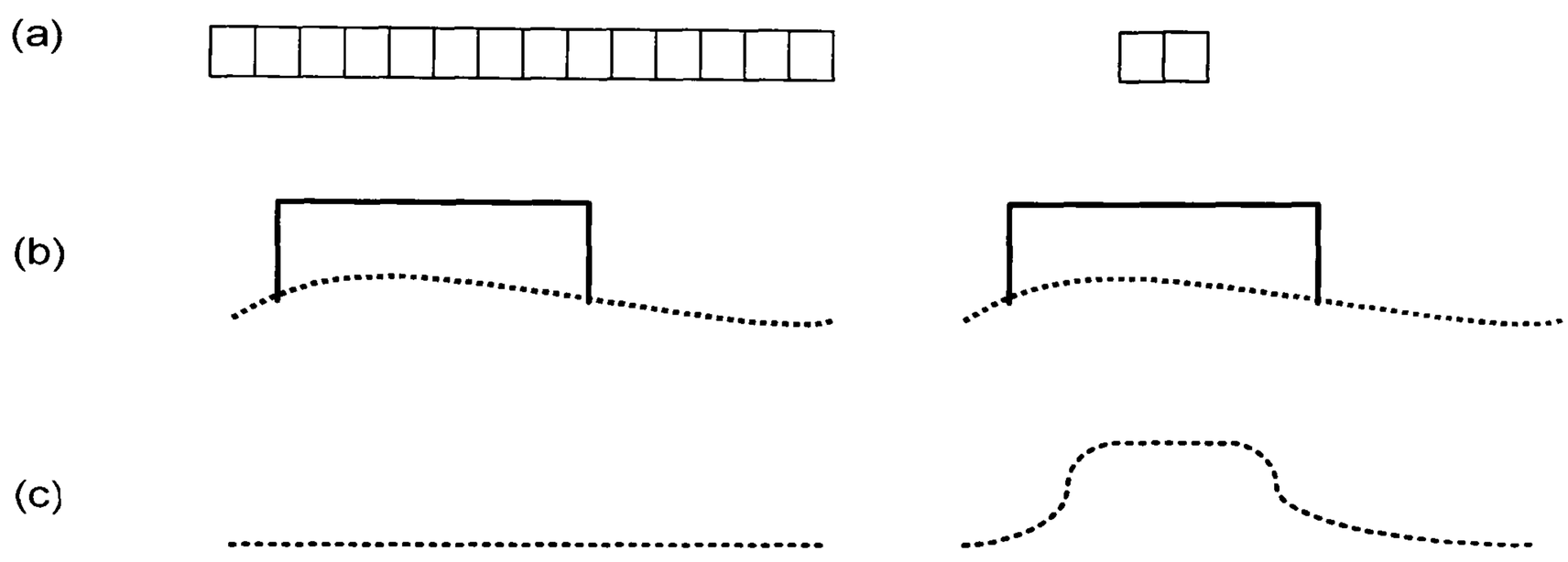


Figure 3-3. Illustration of the "object scale effect". (a) filter in one-dimension; (b) building overlaid with real terrain surface; (c) reconstructed terrain surface by (a)

3.2.3 Adaptive filtering technique

In the previous section, two scale factors, that is the "terrain variation effect" and "object scale effect", have been described as major error sources caused by the deterministic filter, and both effects lead to the generation of Type I and Type II errors respectively. To make the filter robust to both effects, various types of adaptive filtering techniques have been suggested (Axelsson, 1999; Elmqvist, 2001; Kraus & Pfeifer, 1998; Vosselman, 2000; Wack & Wimmer, 2002; Zhang et al., 2002). Unlike the deterministic filter, the adaptive filter adopts the "smoothness prior" for representing natural terrain. The "smoothness prior" assumes generic terrain surface changes smoothly rather than abruptly. This assumption makes the adaptive filter represent more diverse landforms, compared to the "homogeneous prior", which is employed by the deterministic filter. However, the "smoothness prior" of generic terrain surface is generally difficult to define – *a priori*. Generic terrain surface continuously varies in slope or height with different magnitudes. Under this ill-constrained situation, the adaptive filter implicitly defines "favourable" terrain characteristics, by which the filter

derives the characteristics of underlying terrain surface directly from the data given. Thus, the adaptive filter can adjust the filtering criterion to the data-driven terrain characteristics, and the most adequate criterion is locally selected.

Let us define a data-driven terrain characteristic as DT_j, which is extracted from LIDAR points, S_j. The fundamental idea behind the adaptive filter is that a filtering criterion, δ, is determined depending on DT_j. Thus, the labelling function, F, in Eq. 3-2 can be modified as a function of DT_j as follows:

$$f_i = \mathrm{F}\left(s_i \left| \delta_j(DT_j)\right.\right);\ s_i \in S_j \ and\ \ f_i \in \{on, off\} \tag{3-3}$$

Compared to the deterministic filter, there are several advantages in the adaptive filter. As can be seen in Eq. 3-3, the δ_j is not pre-specified as is the deterministic filter, but varies depending on DT. This dynamic ability can allow the adaptive filter to become less sensitive to the "terrain variation effect", and to cope with more complex landforms. In addition, the adaptive filter is robust to the "object scale effect" as the filter does not restrict the size of area where the DT is analyzed, which can make the adaptive filter accommodate various scales of off-terrain objects.

The success of the adaptive filter, however, strongly depends on the reliability of DT extracted; better approximation of DT to a "true" terrain surface is obtained, smaller classification errors of on-terrain and off-terrain points are generated. There are several different ways to obtain the DT and to design an adaptive function of $\delta(DT)$ in Eq. 3-3; DT is semi-automatically collected within pre-specified areas and δ is trained with the analysis of terrain slope information collected from DT (Vosselman, 2000); a statistical median value of slope distribution is used to characterize DT, by which δ is automatically determined according to DT (Axelsson, 2000); DT is predicted by extracting on-terrain points with minimal height within a certain interval, and then more reliable on-terrain points are extracted if height differences between LIDAR points and data-driven DT fall in an expected standard deviation (Pfeifer & Kraus, 1998). The

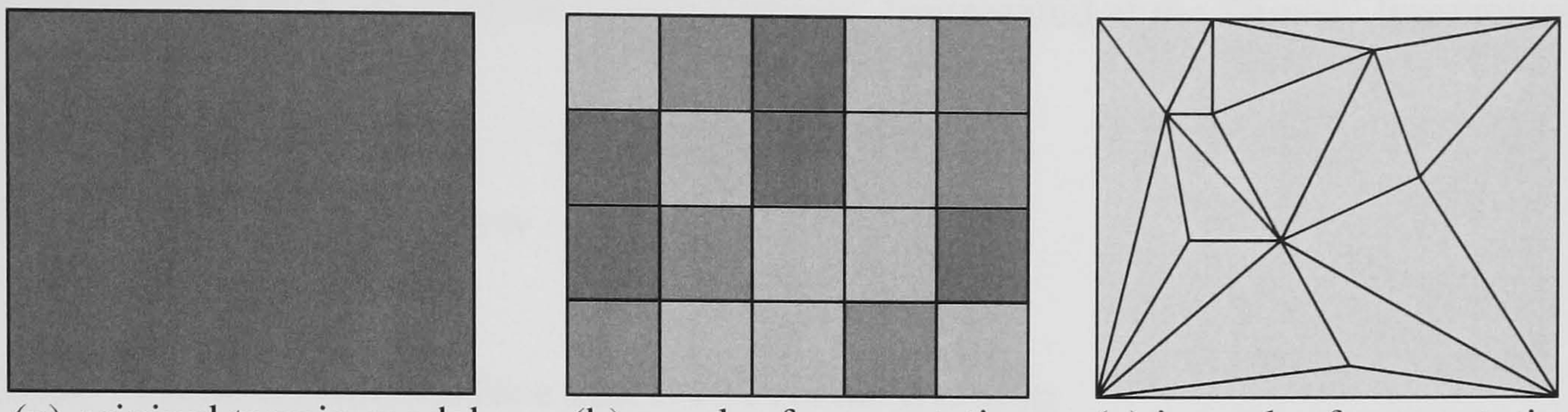
adaptive filters mentioned above have shown that more diverse types of *DT* can be extracted than with the deterministic filters. However, since generic terrain surface is ill-constrained by nature, only *DTs* that are implicitly "favoured" by the filter can be driven, and a number of parameters are required to define the "favourable" terrain characteristics.

3.2.4 New approach

Although the aforementioned adaptive filtering techniques have suggested promising strategies to tackle various landforms, the main problem with the adaptive filters is that different parameters have to be given for various landforms based upon a topographic analysis over investigated areas. This drawback implies that the geometric representation of natural terrain can be formulated only within a limited flexibility in the adaptive filtering mechanism, which leads to the degradation of filtering performance when the filter is applied to areas with "unfavourable" terrain characteristics.

However, a good hypothesis can partially resolve this problem; by assuming that geometric variety of natural terrain would be reduced in smaller regions. A very simple way to achieve this hypothesis is to uniformly fragment the entire data domain into sub-regions with a certain fixed size (see figure 3-4 (b)). Within these sub-regions, the adaptive filtering performance will be improved because terrain characteristics of underlying areas become simplified. However, the problem with the regular fragmentation method is that fragmenting size is arbitrarily determined so that the terrain complexity of each fragmented region is still unpredictable.

To resolve this problem, a better fragmentation method can be adopted, in which a LIDAR DSM, convolved with heterogeneous terrain slopes, is fragmented into a set of homogeneous sub-regions, within which underlying terrain is characterized with a single terrain slope (see figure 3-4 (c)). Once these homogeneous sub-regions are achieved, the separation between on-terrain points and off-terrain points becomes much simpler, as underlying terrain characteristics are uniformly regularized. More importantly, the data-driven terrain characteristic, *DT* , analyzed by a LIDAR filter is not implicitly inferred from an arbitrary dataset, but explicitly modelled as a planar surface. Thus, there is no need for a large number of parameters constraining "favourable" terrain characteristics.



(a) original terrain model (b) regular fragmentation (c) irregular fragmentation

Figure 3-4. Illustration of the suggested terrain fragmentation idea; darker gray colour represent that more different terrain slopes are mixed up in a region.

Based upon the irregular terrain fragmentation idea mentioned above, a new LIDAR filtering technique, called the recursive terrain fragmentation (RTF), was developed. The RTF filter employs an elementary terrain model for the reconstruction of a generic terrain surface. That elementary model is a planar terrain surface, which comprises on-terrain points with the same slope. The RFT filter reconstructs the digital terrain model (DTM) by obtaining plane terrain surfaces hidden in a cloud of LIDAR points. In a similar way to the deterministic filter, a single filtering criterion is adopted in order to differentiate on-terrain points from off-terrain ones, since the terrain favoured by the RTF filter is simply modelled as a plane terrain surface. However, in contrast to the deterministic filter, the RTF filter uses the filtering criterion for observing the "terrain variation effect" (i.e., co-existence of multiple terrain slopes) and verifying a hypothesis modelled as a plane terrain surface over an area. By that, invalid areas, with multiple terrain slopes, are recursively fragmented into sub-regions until all terrain segments fragmented are proven to be plane terrain surfaces. As the "terrain variation effect" is uniformly regularized, the RFT filter can be adjusted to the terrain slope variations.

To realize the RTF concept, a "coarse-to-fine" strategy is adopted for scaling down the fragmentation level from global to local scale. This strategy helps the filter to achieve self-adaption to the "object scale effect", where the fragmentation scale is dynamically determined depending on the size of underlying off-terrain features. In addition, an optimization of the RTF filter is implemented in a "hypothesis-test" framework. The RTF is accomplished over an area as detecting an on-terrain point from a number of candidate points. All plausible fragmentation results are predicted in advance. The

optimization is achieved by selecting the on-terrain point providing the "best" fragmentation result as being plane terrain surfaces. Finally, the DTM can be reconstructed by a set of plane terrain surfaces, fragmented at the "finest" fragmentation scale.

3.3 Method overview

3.3.1 Plane terrain surface and DTM reconstruction

The recursive terrain fragmentation (RTF) filter assumes that a generic terrain surface is a piecewise plane terrain surface. A planar terrain is modelled by PTM (Plane Terrain Model), and the RTF filter reconstructs the digital terrain model (DTM) by recursively obtaining PTMs hidden in a LIDAR DSM. In this context, a DTM, ψ, comprising M plane terrain models is described by

$$\psi = \{\phi_j\}_{j=1}^{M} \tag{3-4}$$

where j is the model index of a PTM, ϕ and $1 \le j \le M$. In Eq. 3-4, a PTM, ϕ_j, represents a planar terrain with single slope. The ϕ_j is modelled by a triangular plane with three vertices, $V_j = \left[v_j^1 \; v_j^2 \; v_j^3 \right]$, which are labelled as on-terrain points. The slope of ϕ_j is referred to as the "characteristic slope", θ_j^c, and represents a terrain slope of underlying area. The θ_j^c is defined as an intersecting angle between a horizontal plane (i.e., z=0) and ϕ_j. Thus, the ϕ_j is defined by three on-terrain points labelled by a filtering function, F, as follows:

$$\forall v_j^k \in V_j \;\; a_j x_j^k + b_j y_j^k + c_j z_j^k + d_j = 0; \;\; F\left(v_j^k \mid \phi_j\left(\theta_j^c\right)\right) = \{on\} \tag{3-5}$$

where $v_j^k = \left[x_j^k \; y_j^k \; z_j^k \right]$ and $1 \le k \le 3$; $\left[a_j \; b_j \; c_j \; d_j \right]$ is the planar surface parameter of ϕ_j; the characteristic slope, θ_j^c, of ϕ_j is given by

$$\theta_j^c = \cos^{-1}\left(\frac{c_j}{\sqrt{a_j^2 + b_j^2 + c_j^2}} \right) \tag{3-6}$$

Now, member points, S_j, which are located inside ϕ_j, can be obtained by investigating following the condition:

$$\forall s_i \in S_j \quad \begin{cases} (v_j^1 - v_j^0) \cdot (s_i - v_j^0) < 0 \\ (v_j^2 - v_j^1) \cdot (s_i - v_j^1) < 0 \\ (v_j^0 - v_j^2) \cdot (s_i - v_j^2) < 0 \end{cases} \tag{3-7}$$

where $\left(v_j^1 \; v_j^2 \; v_j^3 \right)$ are vertices of ϕ_j and s_i is a member point of ϕ_j and $\cdot$ is the dot product of two vectors. Figure 3-5 illustrates the ϕ_j constructed by three on-terrain vertices and member points located within ϕ_j.

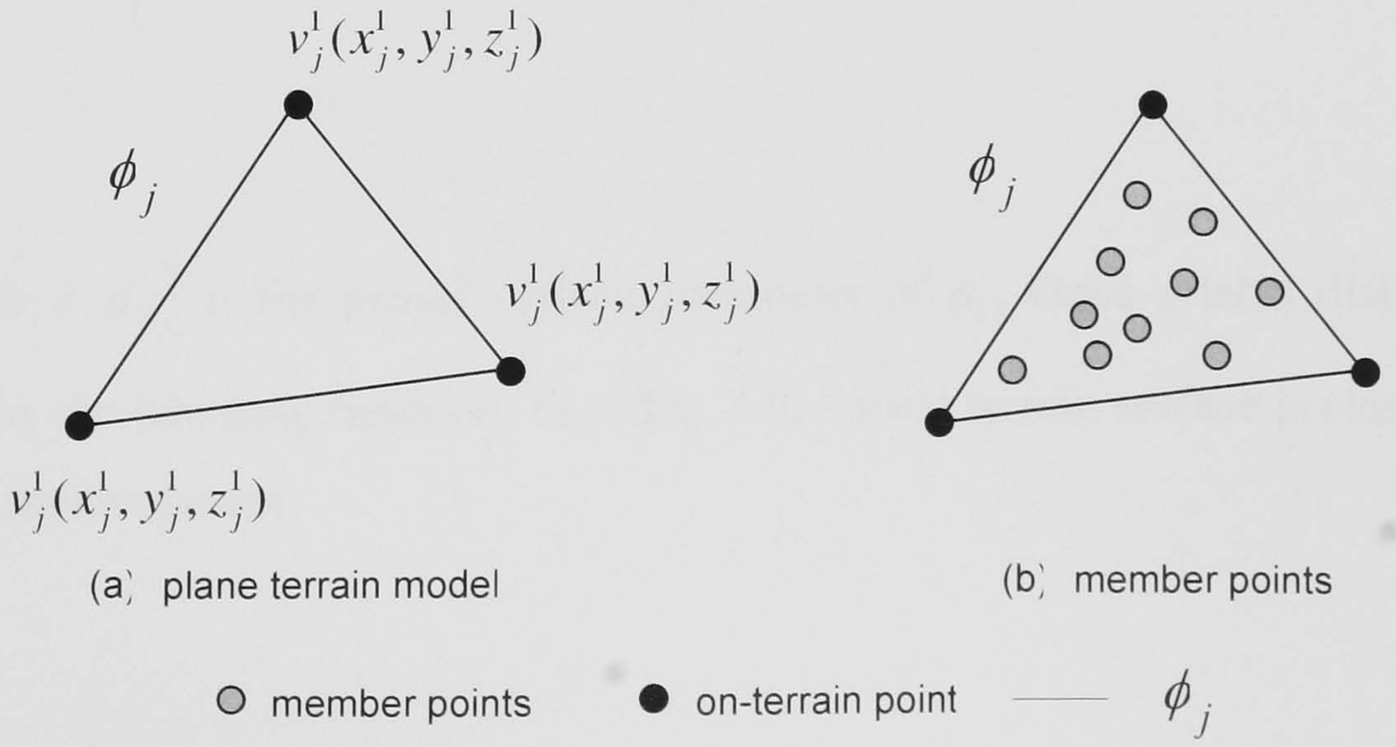


Figure 3-5. Illustration of a plane terrain model and its member points.

3.3.2 Labelling function and terrain classification

Once a local terrain surface is hypothesized as being planar terrain by a PTM, ϕ_j, this hypothesis will be verified only if all the slopes of on-terrain points follow the ϕ_j's characteristic slope, θ_j^c; otherwise it is decided that the different slopes coexist within that area, and the ϕ_j will be fragmented into smaller sub-regions. For this verification process, a labelling function is devised to provide means for detecting on-terrain points and off-terrain points from member points. A label f_i for a member point, $s_i(x_i, y_i, z_i)$ of ϕ_j is determined as one of two labels (i.e., {on}, {off}), by comparing a "geodetic" height between ϕ_j and s_i with a height threshold, δ_h. If a relative height between the height of s_i height and its projection onto the surface of ϕ_j, is less than δ_h, the s_i is labelled as an on-terrain point, otherwise as an off-terrain point. Thus, a labelling function, F, can be described as follows:

$$\forall s_i \in S_j \quad f_i = \mathrm{F}\left(s_i \mid \delta_h, \phi_j\right);\ f_i \in \{on, off\} \tag{3-8}$$

$$\begin{cases} f_i = \{on\} \quad if\ \ z_i - \dfrac{\left(a_j x_i + b_j y_i + d_j\right)}{c_j} \le \delta_h \\ f_i = \{off\} \quad if\ \ z_i - \dfrac{\left(a_j x_i + b_j y_i + d_j\right)}{c_j} > \delta_h; \end{cases} \tag{3-9}$$

where $\left[a_j\ b_j\ c_j\ d_j\right]$ is the planar surface parameter of ϕ_j. Once a label distribution is generated by the labelling function, F, of Eq. 3-9, a local terrain surface is classified into following terrain classes.

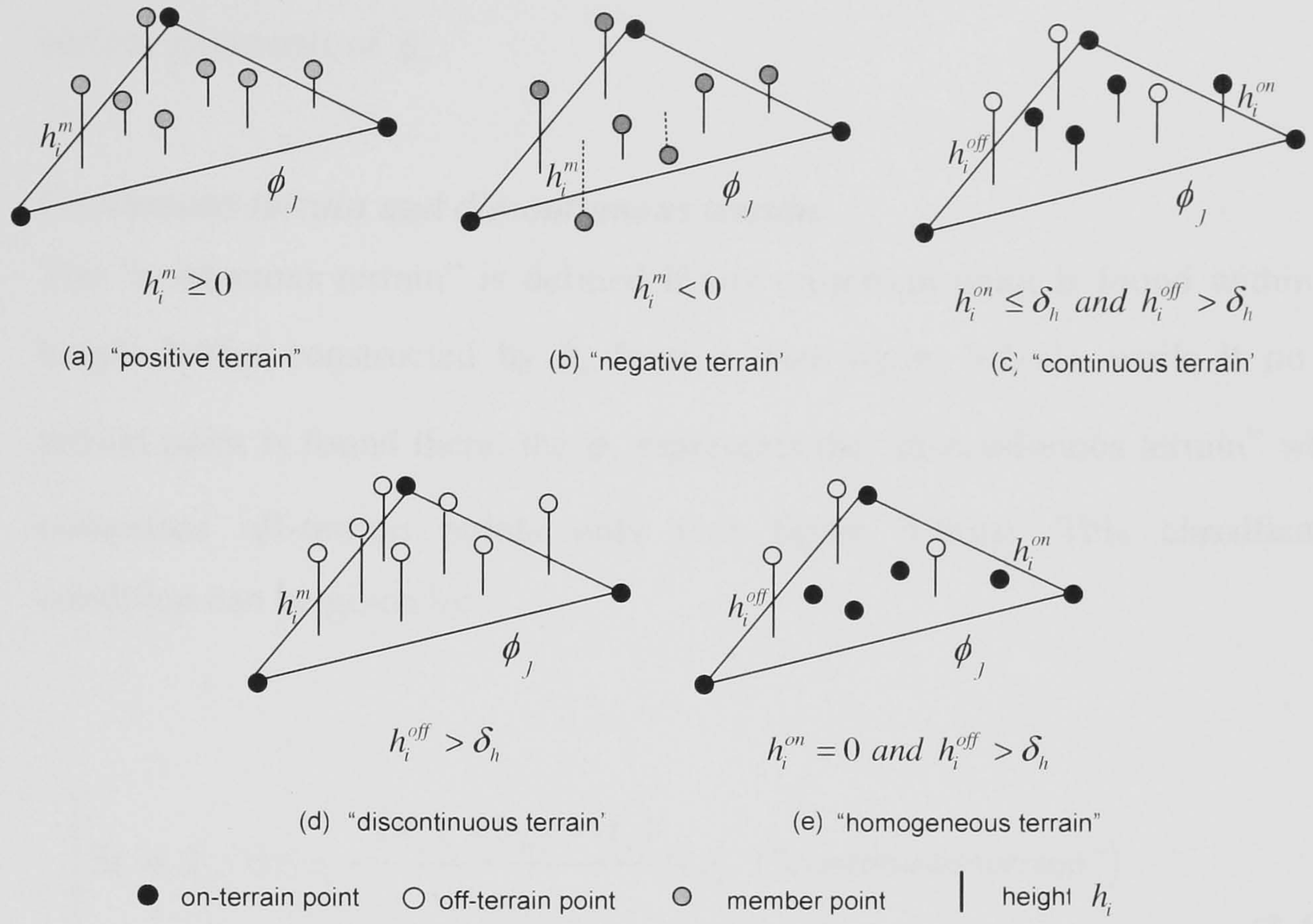


Figure 3-6. Illustration of the plane terrain conditions; h_i^m , h_i^{on} , and h_i^{off} represent the "geodetic" height of a member point, on-terrain, and off-terrain point from the plane ϕ_j , respectively; δ_h is the height criterion

- ***Positive terrain and negative terrain***

  A local terrain surface hypothesized as being planar terrain by a PTM, ϕ_j , will be classified as the "positive terrain" if all the member points are "positively" located above ϕ_j (see figure 3-6(a)), while it will be classified as the "negative terrain" if any member point is located below the ϕ_j (see figure 3-6(b)). This "positive" and "negative" terrain condition can be described by

$$\begin{cases} \forall s_i \in S_j \;\; z_i - \dfrac{\left(a_j x_i + b_j y_i + d_j\right)}{c_j} \geq 0 \;\; (\text{"}positive\,terrain\text{"}) \\ \exists s_i \in S_j \;\; z_i - \dfrac{\left(a_j x_i + b_j y_i + d_j\right)}{c_j} < 0 \;\; (\text{"}negative\,terrain\text{"}) \end{cases} \tag{3-10}$$

where $s_i(x_i, y_i, z_i)$ is a member points of ϕ_j and $\left[a_j\, b_j\, c_j\, d_j\right]$ is the planar surface parameter of ϕ_j.

- ***Continuous terrain and discontinuous terrain***

The "continuous terrain" is defined if any on-terrain point is found within the height buffer constructed by δ_h from ϕ_j (see figure 3-6(c)), while if no on-terrain point is found there, the ϕ_j represents the "discontinuous terrain" which comprises off-terrain points only (see figure 3-6(d)). This classification condition can be given by

$$\begin{cases} \exists s_i \in S_j \quad 0 \le z_i - \dfrac{\left(a_j x_i + b_j y_i + d_j\right)}{c_j} \le \delta_h \quad (\text{"}continuous\ terrain\text{"}) \\ \forall s_i \in S_j \quad z_i - \dfrac{\left(a_j x_i + b_j y_i + d_j\right)}{c_j} > \delta_h \qquad (\text{"}discontinuous\ terrain\text{"}) \end{cases} \tag{3-11}$$

where $s_i(x_i, y_i, z_i)$ is a member points of ϕ_j and $\left[a_j\, b_j\, c_j\, d_j\right]$ is the planar surface parameter of ϕ_j.

- ***Homogeneous terrain***

The "homogeneous terrain" requires that all the slopes of on-terrain points must follow characteristic slope of ϕ_j, θ_j^c. Instead of measuring each angle difference between the ϕ_j and its on-terrain point, s_j^{on}, a relative height between s_j^{on} and ϕ_j is used for determining the "homogeneous terrain", where s_j^{on} must be located on the surface of ϕ_j with no height discrepancy (see figure 3-6(e)). This condition can be described as follows:

$$\forall s_i^{on} \in S_j \quad a_j x_i^{on} + b_j y_i^{on} + c_j z_i^{on} = 0 \tag{3-12}$$

where $s_i^{on}\left(x_i^{on}\ y_i^{on}\ z_i^{on}\right)$ is labelled as an on-terrain point by the labelling function F of Eq. 3-9, and $\left[a_j\ b_j\ c_j\ d_j\right]$ is the planar surface parameter of ϕ_j.

The terrain classification mentioned above serves as a triggering and terminating condition of the terrain fragmentation process. If the ϕ_j is classified as the "homogeneous terrain", or "discontinuous terrain", the ϕ_j is verified as the planar terrain, and the fragmentation is terminated. However, if the ϕ_j is classified as the "negative terrain" or "continuous terrain" which is not verified as the "homogeneous terrain", the ϕ_j is fragmented into smaller sub-regions.

3.3.3 Two-step Divide-and-Conquer Triangulation

The RTF filter recursively fragments a LIDAR DSM from "coarse" to "fine" scale until all the fragmented terrain segments are verified as being planar terrain. This coarse-to-fine terrain fragmentation is implemented by the divide-and-conquer triangulation method. The method recursively decomposes the original problem domain into sub-problems by means of the Delaunay Triangulation. This divide-and-conquer triangulation is formulated with two parts, namely downward and upward terrain fragmentation, depending on the criteria of triggering and terminating the process.

3.3.3.1 Downward terrain fragmentation

The downward terrain fragmentation (DTF) aims to provide an initial terrain model, which will be used in the following upward terrain fragmentation. First, a single planar terrain model is explicitly generated, and hypothesized as a terrain model of the entire LIDAR DSM. Then, the DTF is triggered by investigating the "negative terrain" condition for the initial terrain model. If one or more member points are located below the model surface, a LIDAR point with the minimum distance from the model is selected as on-terrain point, and the model is fragmented into sub-regions by the Delaunay Triangulation. This process continues until all the LIDAR points are located above the terrain models generated.

3.3.3.2 Upward terrain fragmentation

The upward terrain fragmentation (UTF) is the core part of the RTF filter, which leads to the generation of the DTM. The process is similar to the DTF, but the fragmentation is accomplished by selecting on-terrain points in an upward direction. The DTF provides initial terrain models in the form of plane terrain models (PTMs) triangulated with on-terrain points. However, different terrain slopes may coexist in each region hypothesized by the models since the DTF is facilitated by considering the "positive terrain condition" only. The UTF is now focused on removing the "terrain variation effect" from each PTM as more on-terrain points are detected.

Suppose that a certain area is hypothesized as being a plane terrain surface by a PTM, ϕ_j (see figure 3-7(a)). The member points belonging to ϕ_j are labelled as either on-terrain or off-terrain by comparing its relative height from ϕ_j with a height threshold given. The UTF is triggered over ϕ_j if all the on-terrain points do not follow the characteristic slope of ϕ_j as defined in Eq. 3-12. Once the UTF starts over ϕ_j, the most reliable on-terrain point amongst the member points of ϕ_j is selected. With that point, the ϕ_j is fragmented into sub-regions by the Delaunay Triangulation.

The issue is how to obtain the most reliable on-terrain point. To this end, a hypothesis-test optimization method is adopted, in which hypothetical solutions for the upward fragmentation are tested and the "best" hypothesis is determined as an optimum solution. As for generating a hypothesis space, all the on-terrain points generated from ϕ_j are considered as candidate points, which will be used for fragmenting ϕ_j. With three vertices of ϕ_j, each candidate point forms a tetrahedron model, T_j^i (see figure 3-7(b)). As other candidate points are sequentially selected, a hypothesis space for the UTF can be generated for ϕ_j with tetrahedron models, $\left\{T_j^i\right\}$ (see figure 3-7(c)). The three lateral facets of each tetrahedron model can be considered as new PTMs which hypothesize the underlying areas as plane terrain surfaces. Since the new terrain models are introduced over the same region, the relative distances between the member points and corresponding model are changed and thus, the label distributions is altered for each tetrahedron model (see figure 3-7(b) and (c)).

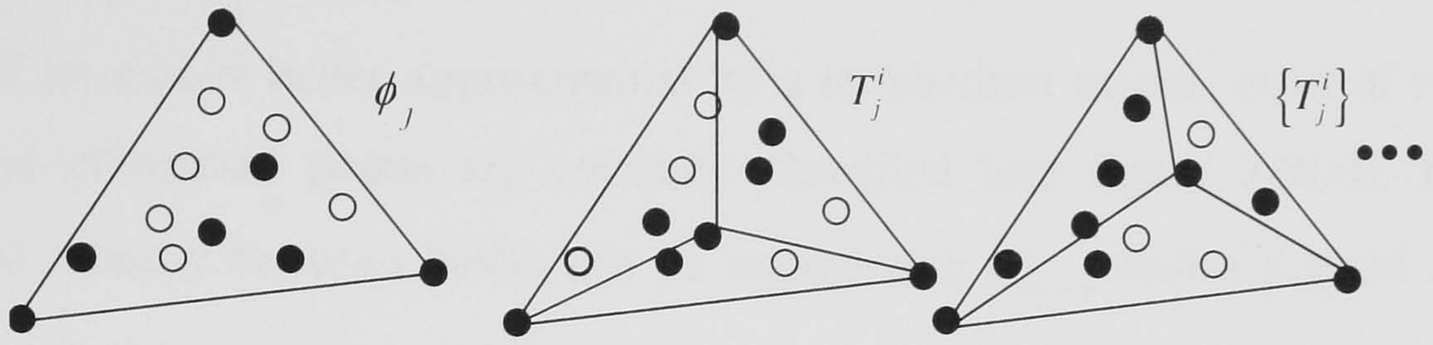


Figure 3-7. Illustration of the upward terrain fragmentation (UTF) process; (a) a plane terrain model, ϕ_j ; (b) and (c) tetrahedron models for the UTF

Now, the problem of detecting the most reliable on-terrain point can be thought of as a model selection problem. The "best" tetrahedron model can be determined as generating the minimum labelling errors. That is, if a "better" tetrahedron model hypothesizes the underlying area as being planar surface, the slope of each facet of the model shall be more closely approximated to the "true" terrain slope. Thus, when on-terrain and off-terrain labels are generated from the model, on-terrain points are more closely located on the model, while off-terrain points are located above the model with an obvious discontinuity. Therefore, the labelling errors of Type I and Type II become minimized as a "better" tetrahedron model is used for the UTF.

The labelling errors caused by a tetrahedron model are measured in terms of a "terrain polarity". As can be shown in figure 3-8, the "terrain polarity" is defined by two angles, where θ_{on-on} ("smoothness polarity") is the angle difference between intra-connected labels and a tetrahedron model and θ_{on-off} ("discontinuity polarity") is the angle difference between inter-connected labels and a tetrahedron model.

Figure 3-8. Definition of "terrain polarity"; (a) "smoothness polarity" and (b) "discontinuity polarity".

A desirable "terrain polarity" is measured for a "better" tetrahedron model as θ_{on-on} gets closer to 0°, while θ_{on-off} gets closer to 90° (see figure 3-9). That is, if a real terrain surface of an area is better approximated by a tetrahedron model, most of real terrain points and off-terrain points are correctly classified (see figure 3-9(a)). By that, a contextual relation between labels can be augmented; θ_{on-on} shows a good agreement with the underlying terrain model's characteristic slope, while a discontinuity of θ_{on-off} is enhanced (see figure 3-9(b)). In consequence, two peaks of "terrain polarity" appear closer to 0° and 90°; otherwise, the distance between two peaks becomes closer and a Gaussian distribution of "terrain polarity" can be observed since the labelling errors become larger (see figure 3-9 (b) and (d)). Therefore, the "terrain polarity" measurement serves as a criterion for the selection of the "best" tetrahedron model, which shows the most strongly polarized distribution of the "terrain polarity" measurement.

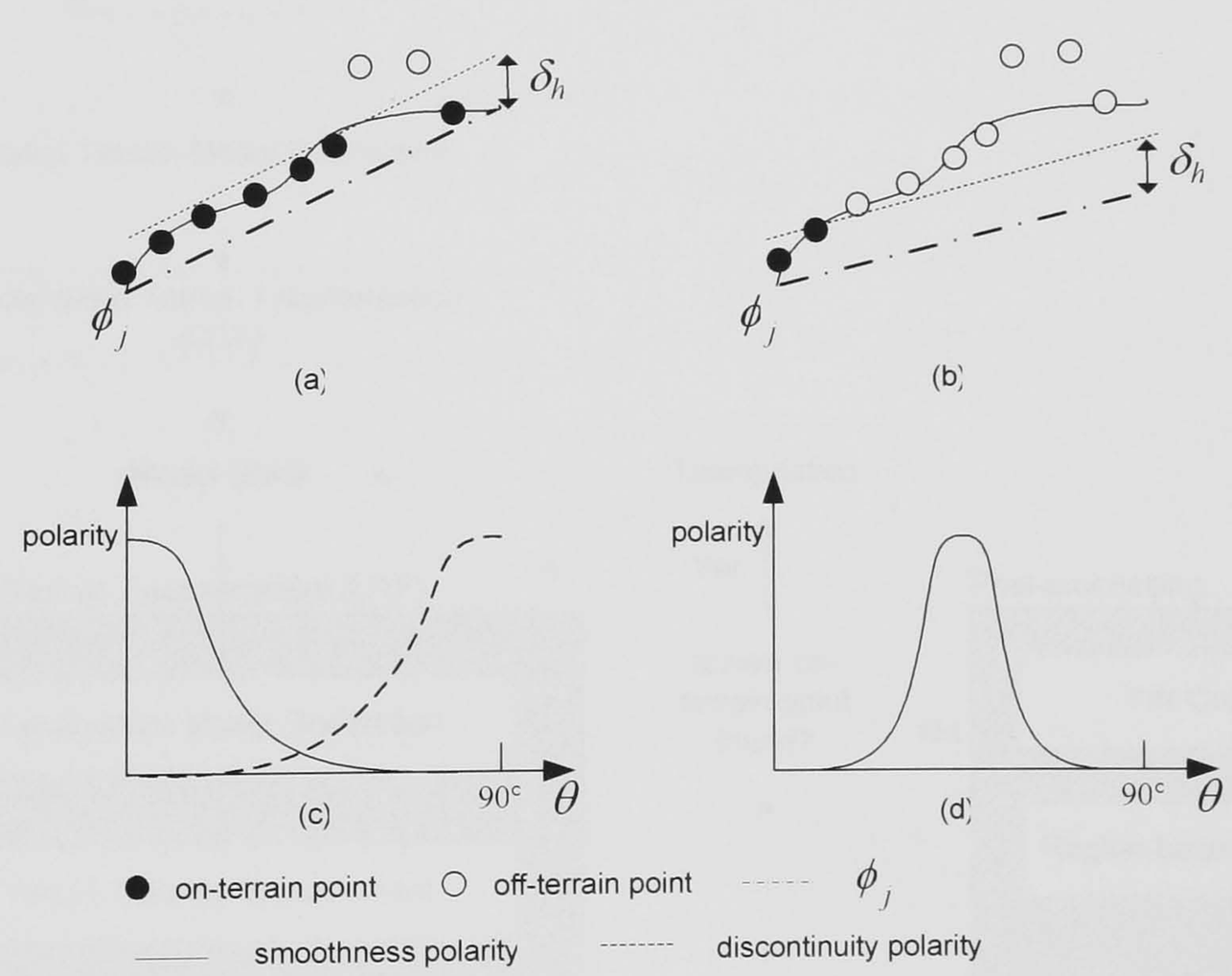


Figure 3-9. Illustration of "terrain polarity" measurement; (a) and (c) illustrate a label generation and terrain polarity distribution by "better" terrain model; (b) and (d) illustrates a label generation and terrain polarity distribution by "worse" terrain model

This model selection is implemented in MDL (Minimum Description Length) framework, where the "terrain polarity" for each tetrahedron candidate is converted into

an information length, and a model with the shortest length is determined as the best model. Finally, by selecting the on-terrain candidate points constructing the "best" tetrahedron model, the current PTM, ϕ_j, is fragmented into smaller sub-regions by the Delaunay Triangulation. This process continues until all the PTMs generated are validated as plane terrain surfaces with single slopes. At the finest fragmentation level the DTM can be generated

3.4 Recursive terrain fragmentation filter for DTM generation

Figure 3-10 shows the overall process of the developed recursive terrain fragmentation (RTF) filter developed. The process can be formulated into following categories; i) initial terrain model generation, ii) downward terrain fragmentation (DTF), iii) upward terrain fragmentation (UTF), and iv) post-processing. In this section, each processing module will be described in more detail.

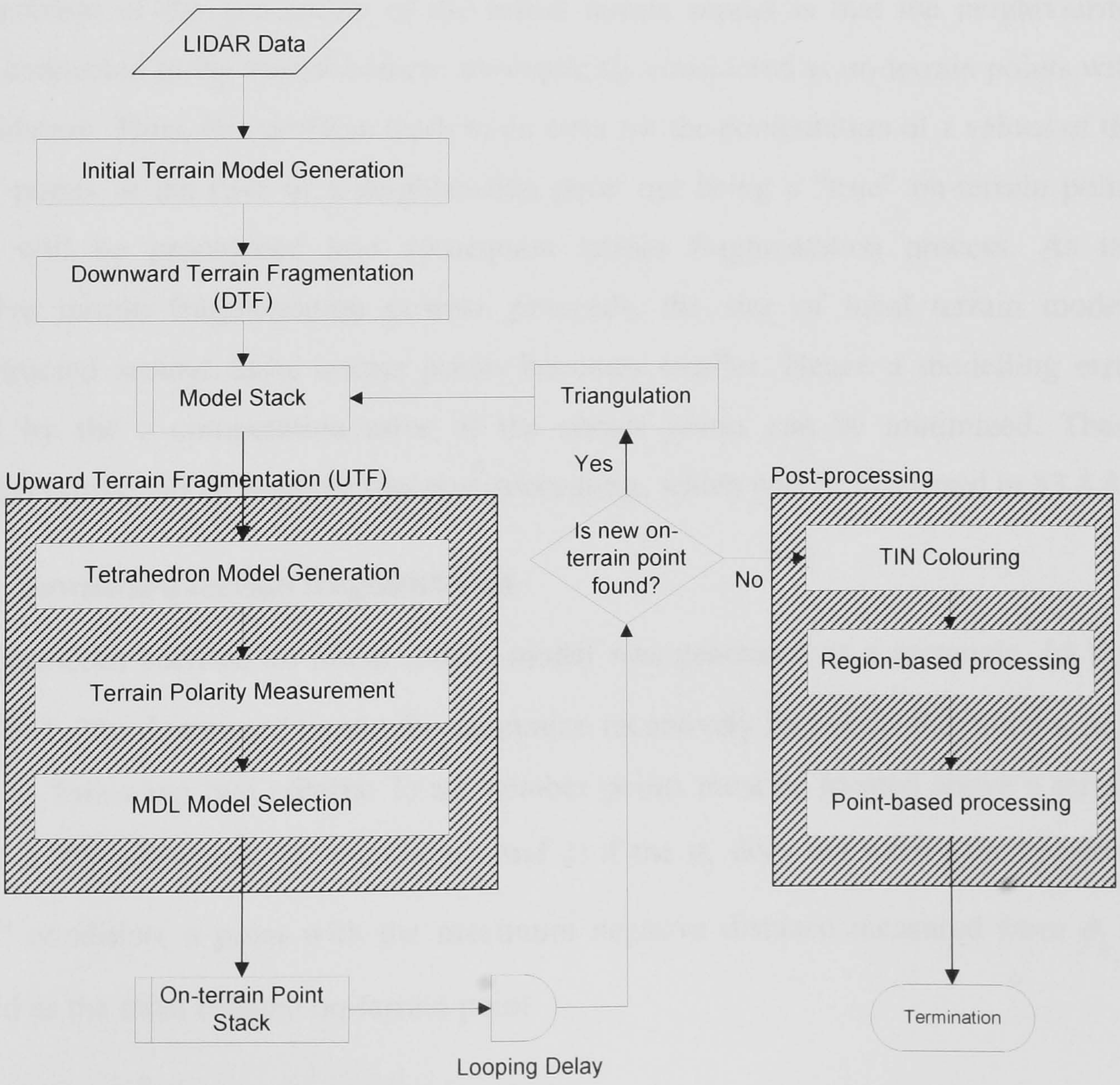


Figure 3-10. Overall process of RTF filter developed

3.4.1 Initial terrain model preparation

At the coarsest level of the terrain fragmentation process, a terrain surface model, ψ, is initialized with a rectangle which covers the entire LIDAR dataset, S. The terrain characteristic of S is hypothesized as a plane terrain surface by the initial terrain model. The initial terrain model is generated by the use of the domain information of S. Four corner points are virtually generated to construct the initial terrain model with a rectangular shape, and x and y coordinates for those virtual corners are determined by computing the minimum and maximum information of (x, y) from S. In order to assign z information to the virtual corner points, the virtual corner points are added to the point list of S and a TIN is constructed. Then, z values of neighbouring points connected to each corner point are collected and the minimum z value is assigned as the height for each corresponding corner point. Finally, these corner points are labelled as on-terrain points, and the initial terrain model is prepared (see the top of figure 3-11).

One problem of the generation of the initial terrain model is that the neighbouring points connected to the virtual corners are explicitly considered as on-terrain points with no validation. Thus, this problem leads to an error for the computation of z values of the corner points in the case of a neighbouring point not being a "true" on-terrain point, which will be propagated into subsequent terrain fragmentation process. As the recursive terrain fragmentation process proceeds, the size of local terrain models reconstructed around those corner points becomes smaller. Hence a modelling error caused by the z computation error of the corner points can be minimized. These localized errors will be corrected by post-processing, which will be discussed in §3.4.4.

3.4.2 Downward terrain fragmentation

In the previous section, an initial terrain model was generated as a rectangle, $\{\phi_j\}_{j=1}^{k}$, where k=1. The downward terrain fragmentation recursively fragments the initial terrain model by following two criteria; 1) all member points must be located above a terrain model, ϕ_j ("positive terrain" condition), and 2) if the ϕ_j does not satisfy the "positive terrain" condition, a point with the maximum negative distance measured from ϕ_j is selected as the most reliable on-terrain point.

The member points belonging to ϕ_j are obtained by Eq. 3-7 and the "positive terrain" condition is investigated by Eq. 3-10. If a negative point located below ϕ_j is found, its distance is measured from ϕ_j and stored in a sequential data list. When this process is completed over ϕ_j, a point with the maximal negative distance is selected from the sequential data list and the point is labelled as on-terrain point. This investigation process, to search for the negative points, is performed over an entire model space of $\{\phi_j\}$ generated at a certain terrain fragmentation level. Then, a new TIN is constructed by using these newly found on-terrain points and the ones generated at a previous fragmentation level. All the triangles constructed in the TIN are fed into the next level of DTF as plane terrain models. Hence, the dimension k of the reconstructed terrain model increases. This process continues until no negative point is found within currently reconstructed terrain model, $\{\phi_j\}$ (see figure 3-11).

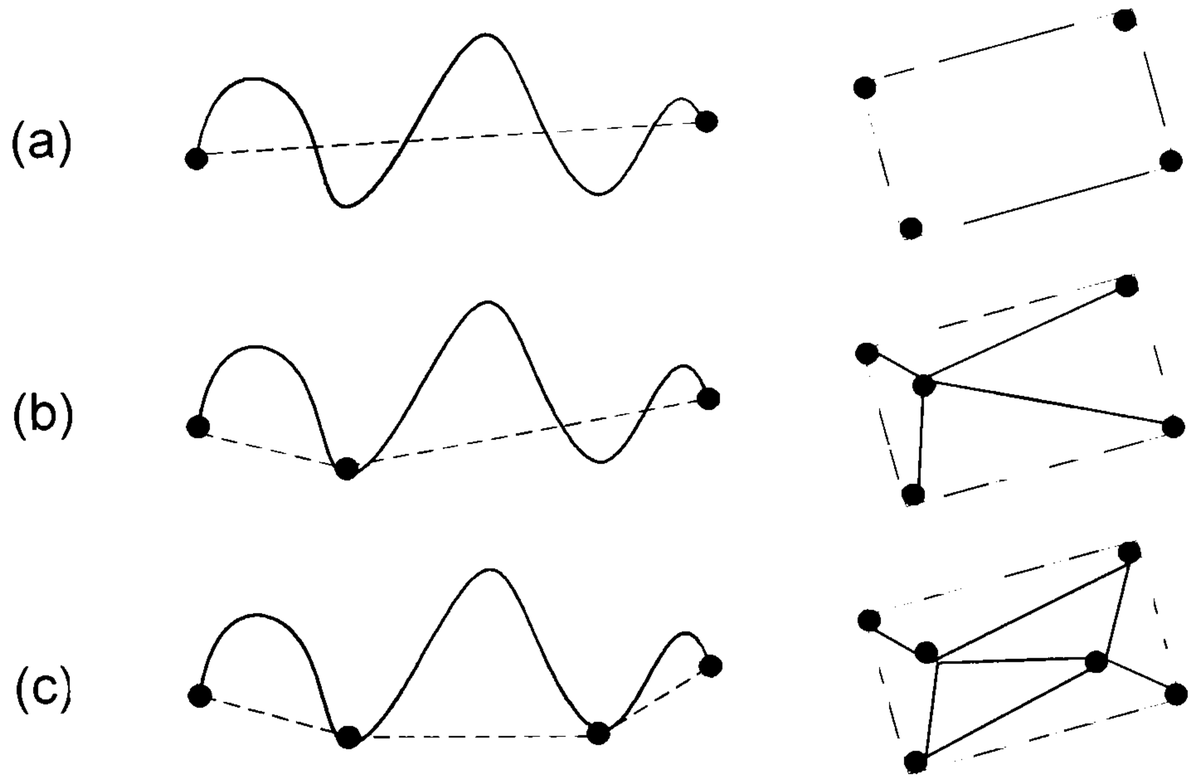


Figure 3-11. Illustration of the downward terrain fragmentation (DTF) process; (a) initial plane terrain model, (b) intermediate result of the DTF, and (c) final result of the DTF

3.4.3 Upward terrain fragmentation

In §3.3.3.2, the overall concept of the upward terrain fragmentation was presented. The upward terrain fragmentation aims to find a local terrain surface with multiple terrain slopes, and partitions that area into terrain segments, which are characterized as a planar terrain with single slope. In UTF, the fragmentation is accomplished by selecting an on-terrain point from a target area with heterogeneous slopes. With possible fragmenting

hypotheses, a point, leading to the minimum classification errors is selected as the best solution of UTF, and thus, the global performance of UTF can be optimized. In a coarse-to-fine strategy, the UTF is recursively performed, and results in the generation of DTM.

3.4.3.1 Triggering and terminating condition

A set of plane terrain model, $\{\phi_j\}_{j=1}^{k}$, is generated through the downward terrain fragmentation. These local terrain models are stored in a "model stack" as a form of TIN. In principle, the upward terrain fragmentation is triggered over an area with multiple terrain slopes, while it is terminated for a plane terrain with single slope. Thus, the UTF must investigate whether the underlying terrain of a PTM can be proven as being a plane terrain surface.

Suppose that a local terrain model, ϕ_j, is selected from the "model stack", and ϕ_j occupies a certain terrain area, where the terrain characteristic is unknown. As discussed in §3.3.3.2, the validation as being a plane terrain surface is determined by classifying the local terrain surface into five different classes; "positive terrain", "negative terrain", "continuous terrain", "continuous terrain", and "homogeneous terrain". Figure 3-12 illustrates the plane terrain validation procedure. First, a "positive terrain" condition for ϕ_j is investigated by Eq. 3-10. To this end, member points belonging to ϕ_j are obtained by Eq. 3-7. The "geodetic" distances between the member points and ϕ_j are calculated as relative heights of member points. If any member point is located below ϕ_j so that its relative height has negative value, the UTF is terminated for ϕ_j, but the fragmentation of ϕ_j will be done by the DTF as presented in §3.4.3. However, if all the member points are located above ϕ_j so that ϕ_j is validated as the "positive terrain", determining the upward terrain fragmentation of ϕ_j moves to the next investigation process.

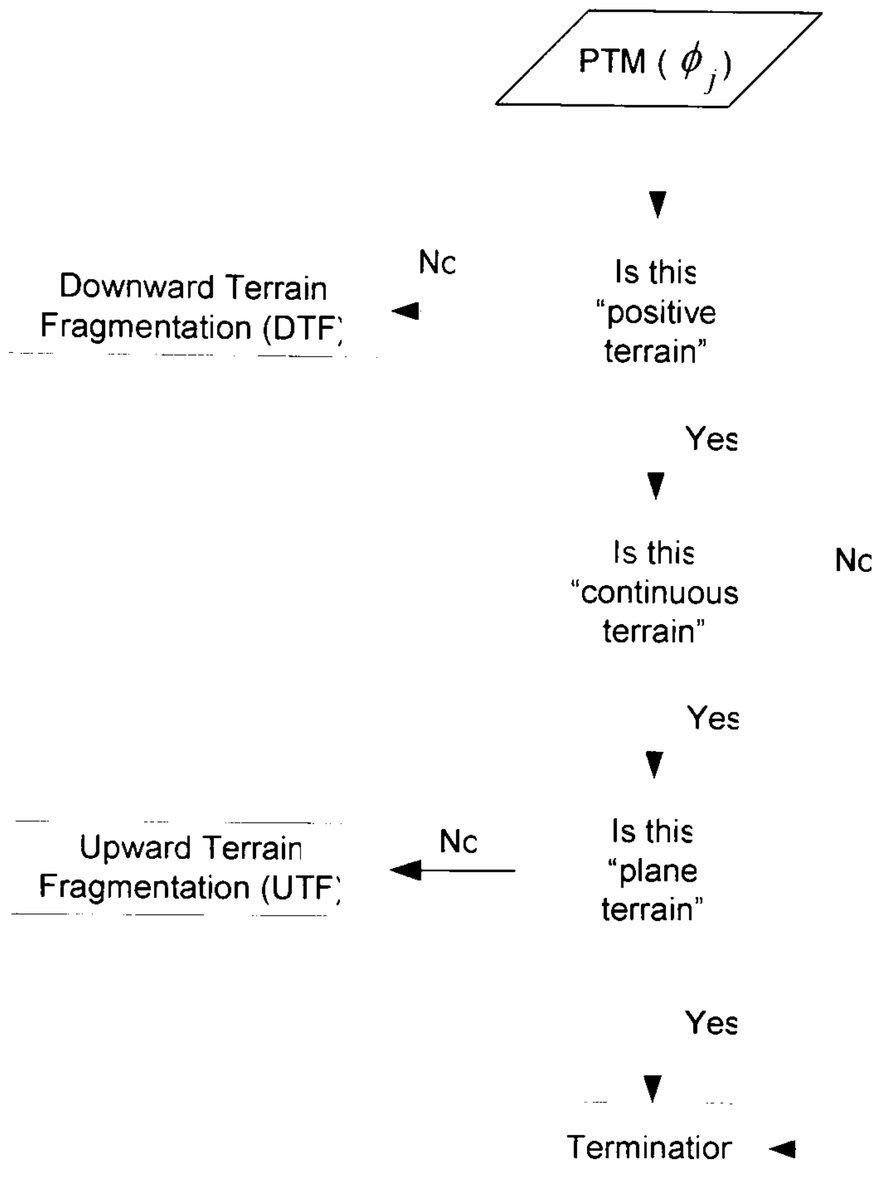


Figure 3-12. Illustration of the plane terrain validation procedure

The next investigation process is given by the "continuous terrain" condition of Eq. 3-11. By investigating this condition, the "positive terrain" is classified as "continuous terrain" or "discontinuous terrain". The member points of ϕ_j are labelled as either on-terrain or off-terrain point by Eq. 3-9; a member point whose relative height is less than a pre-defined height threshold, δ_h, is labelled as an on-terrain point, otherwise, it is labelled as an off-terrain point. With the labels generated, ϕ_j is determined as the "discontinuous terrain" if no on-terrain point can be populated from ϕ_j and the UTF is terminated for ϕ_j (see figure 3-13 (a)). If any on-terrain point can be generated from ϕ_j, the ϕ_j is classified as the "continuous terrain" and further validation process is required over ϕ_j.

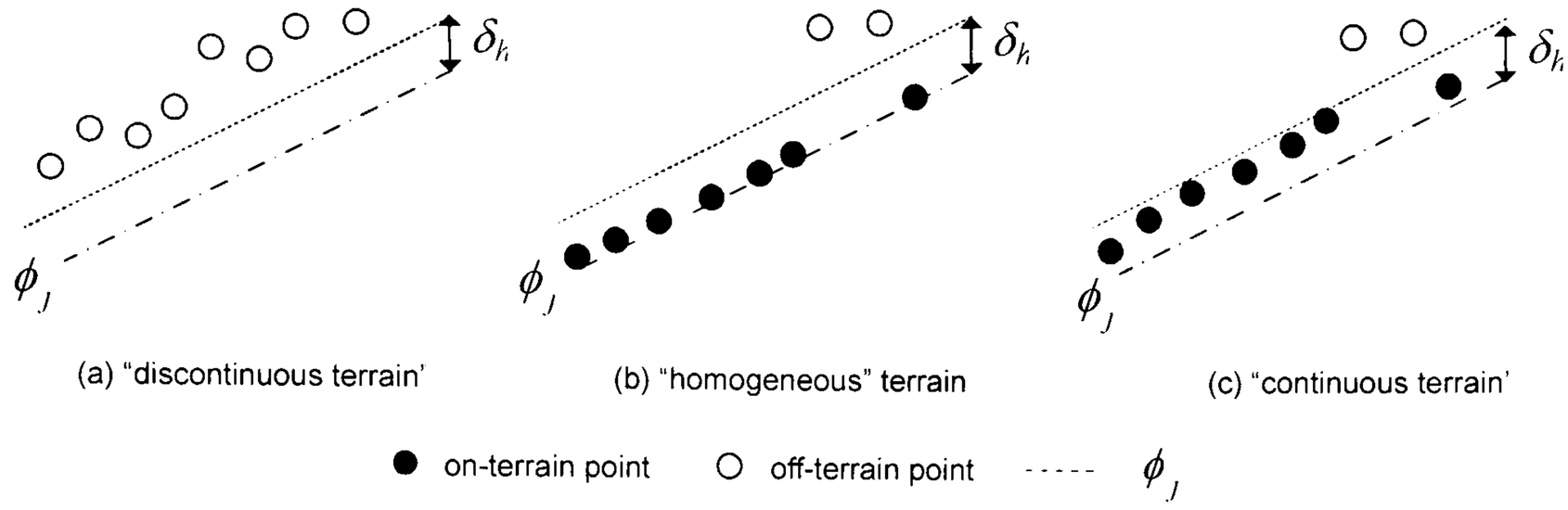


Figure 3-13. Illustration of a triggering and terminating condition of the upward terrain fragmentation.

The final validation of ϕ_j as being a plane terrain surface is determined by investigating the "homogeneous terrain" condition defined in Eq. 3-11. If all the on-terrain points of ϕ_j are located on ϕ_j with no height discontinuity, the UTF is terminated for ϕ_j (see figure 3-13 (b)). However, the UTF is triggered for ϕ_j in case any on-terrain point generated from ϕ_j is located above ϕ_j with a height discontinuity (see figure 3-13 (c)).

In summary, given a plane terrain model, ϕ_j, occupying certain area, the UTF is triggered only if LIDAR points exist within a height buffer space, which is made between ϕ_j and δ_h, and all these points are located above ϕ_j with a height discontinuity ("continuous terrain"). In other cases where either only off-terrain points exist over ϕ_j ("discontinuous terrain") or on-terrain points are located ϕ_j with no height difference ("homogeneous terrain"), the UTF is terminated for ϕ_j and a next local terrain model, ϕ_{j+1}, is selected from the "model stack". The aforementioned investigation process is applied to ϕ_{j+1}.

The selection of δ_h that determines the size of the height buffer is important to trigger or terminate UTF, which affects the final result of RTF filter. There are two factors considered for the selection of δ_h. On the one hand, the δ_h is a vertical height discontinuity to define off-terrain features from a flat or plane terrain surface. The selection of δ_h is very simple and intuitive. One can select a δ_h of 1 metre to differentiate trees or buildings from the street, assuming that trees and building are

higher than δ_h. The problem is that a low object, less than 1 metre, such as small cars or bushes may be classified as on-terrain points (see figure 3-14(a)). On the other hand, the δ_h must be sensitive to detect the "terrain variation effect". As can be shown in figure 3-14(b), the coexistence of different terrain slopes can be found only if on-terrain points are detected within a height buffer constructed by δ_h. A higher value of δ_h can make the filter adaptive to wider range of terrain variation, but at the price of involving more off-terrain features in the terrain model reconstructed. Thus, there is a conflict between the aforementioned two factors for selecting δ_h. A "good" compromise can be achieved as the point density of the LIDAR data increases. Although a steep terrain produces a smaller intersecting area with the height buffer constructed by a lower δ_h, LIDAR points could be acquired over the intersecting region if the point density becomes higher. For instance, given the LIDAR data with a moderate point spacing of one point per 1 metre, δ_h of 1 metre allows the filter to detect terrain slope variations up to 45°, which can cope with most natural terrains. Since today's LIDAR systems provide more than 1 metre point spacing data, δ_h can be lowered so that fewer off-terrain artifacts remain in the DTM reconstructed. Even if low off-terrain features remain, they can be easily removed by a simple post-processing algorithm such as the morphological filter as the size of low off-terrain features remaining is normally very small. This problem will be discussed more in chapter 4.

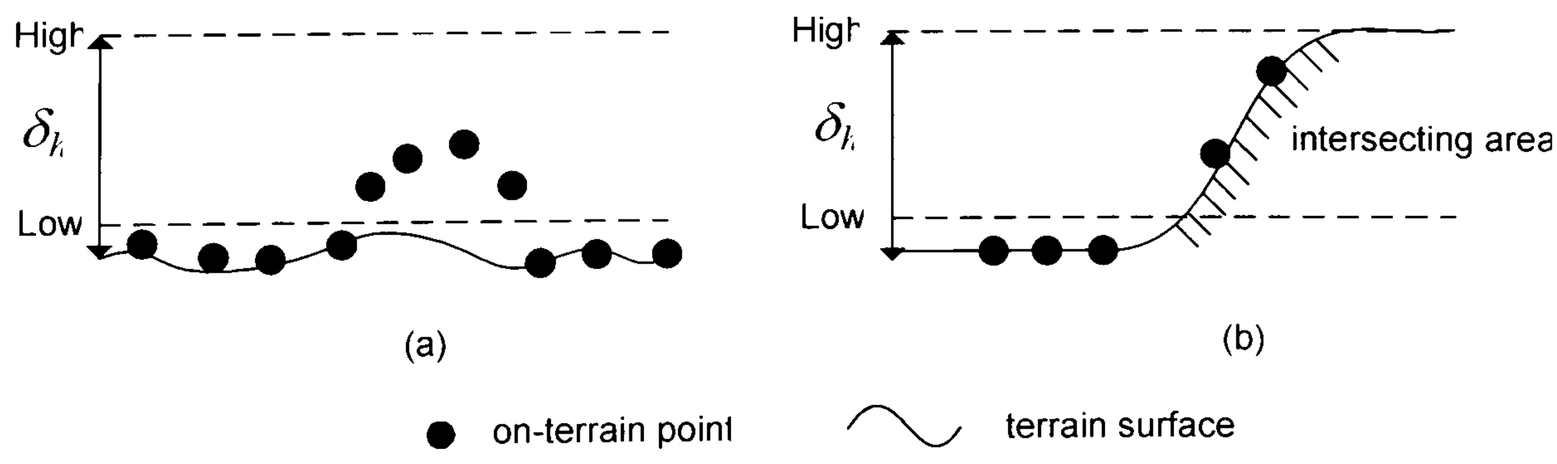


Figure 3-14. Illustration of the selection of δ_h

3.4.3.2 Tetrahedron model generation

Once the UTF is triggered for a plane terrain model, ϕ_j, the process ends up with obtaining a new on-terrain point. For the selection of that point, a number of member

points are considered as candidates for the new on-terrain point selected for UTF. Thus, possible fragmentation results of ϕ_j are predicted before an actual fragmentation is determined which leads to the generation of a hypothetical solution space for UTF.

For generating the hypothesis space, on-terrain points generated from ϕ_j with a height threshold, δ_h, are used as the candidate points for UTF. With three vertices of ϕ_j, each candidate point constructs a tetrahedron model, T_j^i and its three lateral facets, i.e. $\phi_j^{i_k}$ where $1 \le k \le 3$, are hypothesized as new PTMs of underlying area, which would be an actual fragmentation result of ϕ_j (see figure 3-15(a)). Note that a tetrahedron model, T_j^i, is rejected as a hypothesis of UTF if all the three lateral facets of T_j^i are not proven as a "positive terrain" defined by Eq. 3-10. Considering all the candidate points, a number of tetrahedron model, $\{T_j^i\}$, are generated as a hypothesis space of UTF for fragmenting ϕ_j (see figure 3-15(b) and (c)).

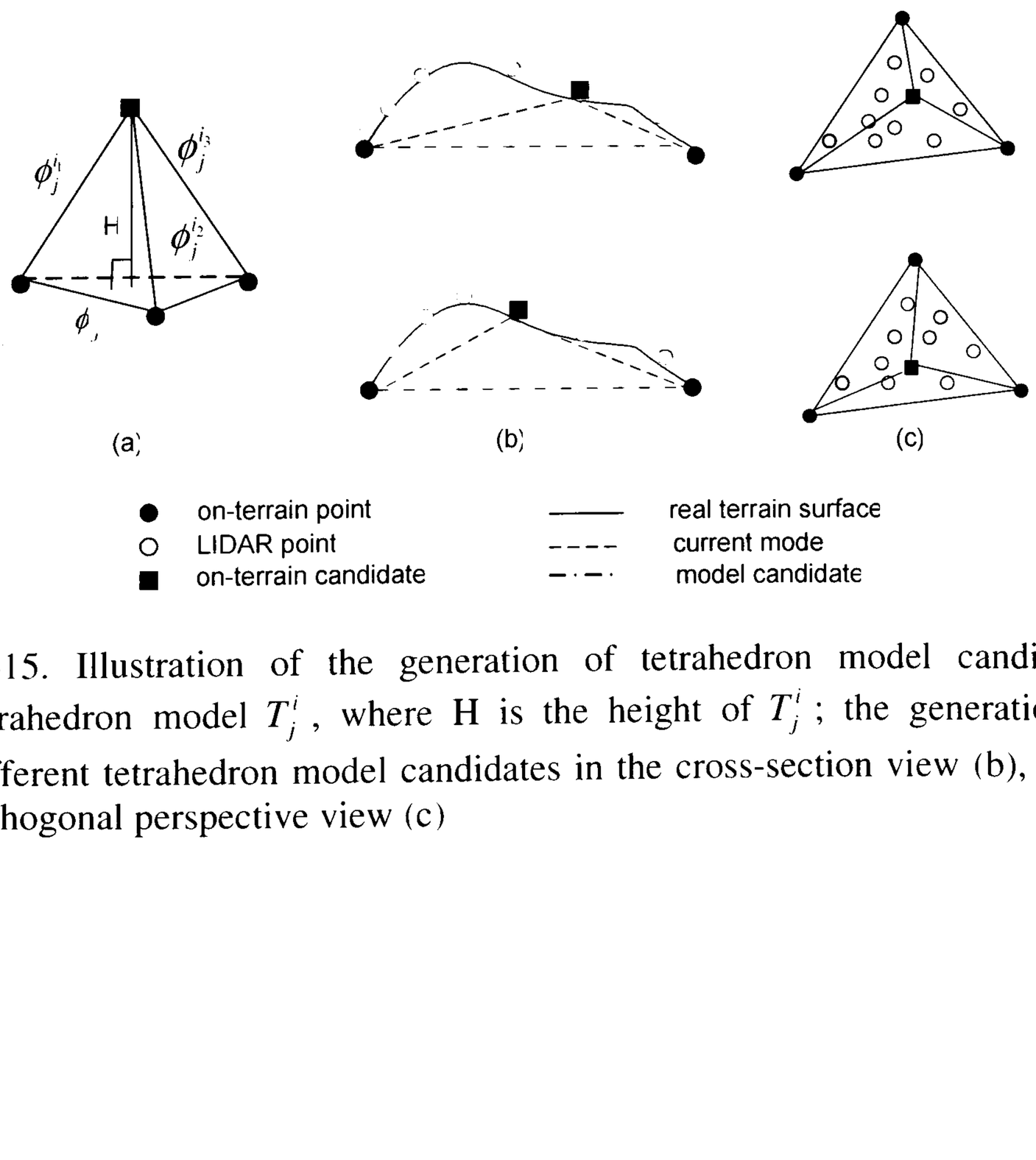


Figure 3-15. Illustration of the generation of tetrahedron model candidates; (a) tetrahedron model T_j^i, where H is the height of T_j^i; the generation of two different tetrahedron model candidates in the cross-section view (b), and in the orthogonal perspective view (c)

3.4.3.3 Terrain polarity measurement

In the UTF, the best hypothesis to fragment a PTM, ϕ_j, is determined after all the hypotheses represented by a set of tetrahedron models, $\{T_j^i\}$, are tested in terms of "terrain polarity". As discussed in §3.3.3.2, "terrain polarity" provides a tool to quantify the labelling errors caused when an area is hypothesized as a plane terrain surface by a PTM. This labelling error is measured, based on a contextual relation between labels. Two contextual relationships are defined such as an intra-connection and inter-connection. The intra-connection is a relationship where a point and its neighbouring points are connected with the same labels; while the inter-connection is defined as a relation connected with different labels. By these two different label connections, a new label set is generated for T_j^i in order to measure the "terrain polarity".

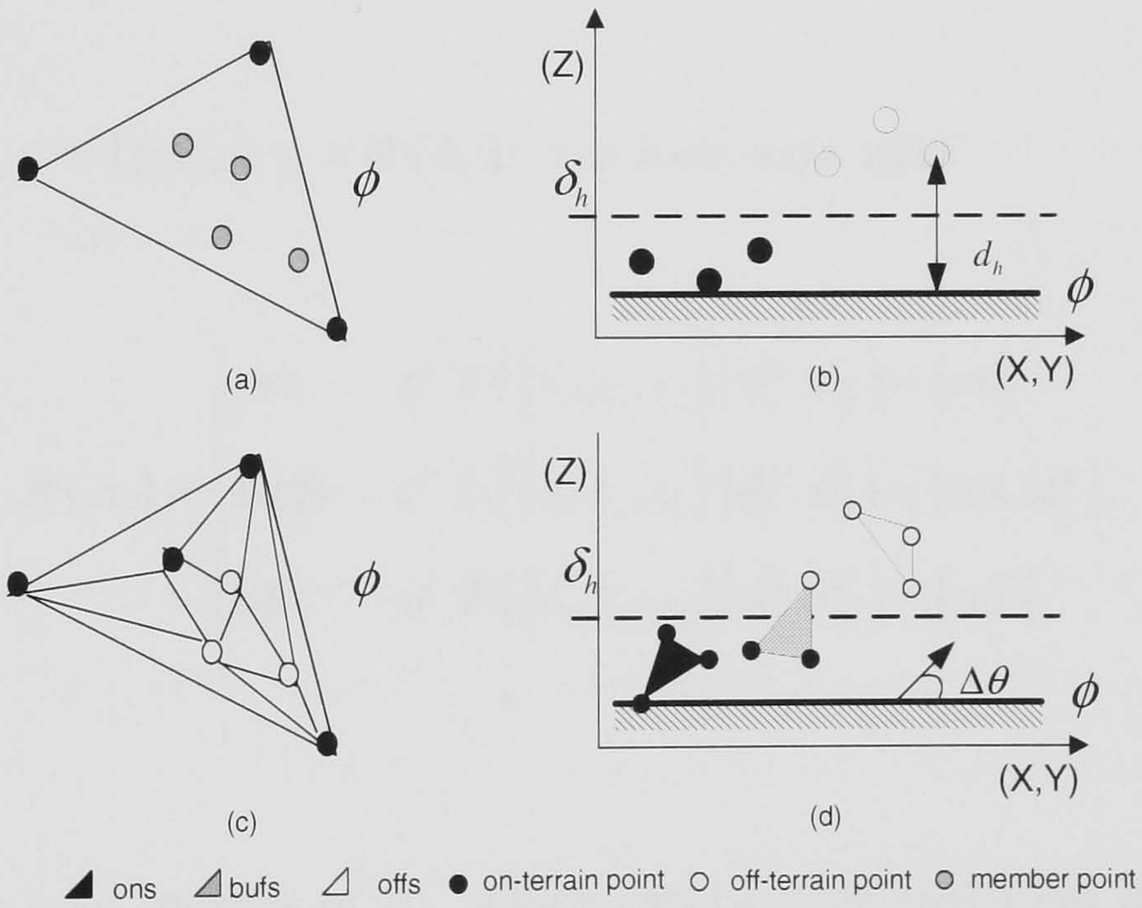


Figure 3-16. Illustration of observation model used for the polarity measurement

Suppose a tetrahedron model, T_j^i is generated by selecting an on-terrain candidate point of ϕ_j and T_j^i has three lateral facets such as $\phi_j^{i_k}$ where $1 \le k \le 3$. Figure 3-16(a) shows one of the three facets, $\phi_j^{i_k}$, and its member points obtained by Eq. 3-7. By computing the relative heights of the member points from $\phi_j^{i_k}$, their labels are determined as either

on-terrain or off-terrain label when a height threshold, δ_h, is given by Eq. 3-9 (see figure 3-16 (b)).

In order to obtain the aforementioned inter- and intra-connection between on-terrain and off-terrain labels, a TIN is constructed over $\phi_j^{i_k}$ with three vertices of $\phi_j^{i_k}$ and its member points (see figure 3-16 (c)). Based on this, a new label set is defined as {ons, buffs, offs}, from which a new label, γ_i, is assigned to each triangle, Δ_i, of the TIN generated from $\phi_j^{i_k}$'s member points. Here, a {ons} is assigned to Δ_i when all the three vertices of Δ_i are labelled as on-terrain points; a {buffs} is assigned when Δ_i is formed with a mixture of on- and off-terrain labels; a {offs} is assigned to Δ_i if all the vertices of Δ_i are labelled as off-terrain points. Therefore, a new labelling function, R, can be given by

$$\gamma = \{\gamma_i\}_{i=1}^{N};\ \gamma_i = \mathrm{R}(\Delta_i);\ \gamma_i \in \{ons, bufs, offs\} \tag{3-13}$$

$$R(\Delta_i) = \begin{cases} ons & if\ F\left(\{s_i, s_j, s_k\} \mid \phi_j^{i_k}, \delta_k\right) = \{on\} \\ buffs & if\ F\left(\{s_i, s_j, s_k\} \mid \phi_j^{i_k}, \delta_k\right) = \{on, off\} \\ offs & if\ F\left(\{s_i, s_j, s_k\} \mid \phi_j^{i_k}, \delta_k\right) = \{off\} \end{cases} \tag{3-14}$$

where N is the total number of Δ_i, which can be generated from $\phi_j^{i_k}$; $\{s_i, s_j, s_k, \}$ are vertices of Δ_i; F is the labelling function to assign on-terrain or off-terrain to a single LIDAR point by a height threshold, δ_h, defined in Eq. 3-9.

Now, the angle difference, $\Delta\theta$, between $\phi_j^{i_k}$ and an arbitrary label, γ, is calculated for measuring the "terrain polarity" as follows:

$$\Delta\theta = \cos^{-1}\left(\frac{a_1 a_2 + b_1 b_2 + c_1 c_2}{\sqrt{a_1^2 + b_1^2 + c_1^2}\sqrt{a_2^2 + b_2^2 + c_2^2}}\right) \tag{3-15}$$

where a plane through three vertices of $\phi_j^{i_k}$ is defined as $a_1 x + b_1 y + c_1 z + d_1 = 0$ and the other plane through three vertices of Δ_i is defined as $a_2 x + b_2 y + c_2 z + d_2 = 0$. In the following section, the angle difference, $\Delta\theta$, serves as a parameter to determine a degree of the "smoothness" and "discontinuity" polarity depending on a label assigned to Δ_i.

3.4.3.4 Model selection by Minimum Description Length criterion

As discussed in §3.4.3.2, a set of tetrahedron models, $\{T_j\}$, generated for a plane terrain model (PTM), ϕ_j, represents potential fragmentation solutions for an area occupied by ϕ_j. Given the tetrahedron models and the LIDAR data, the UTF is completed as selecting the model which leads to best fragmentation result. To accomplish this model selection task, a criterion for the "goodness of a model" is required. In general, residuals between a model and the data would be used to measure the "goodness of fit", where the model leading to a minimum residual is selected as the best model. However, the residual cannot be adopted for the current purpose, since the residual error is not simply modelled as a white noise following the Gaussian normal distribution. For instance, off-terrain points are located above the model surface with no constraint. In this case, the model generating the minimum residual error would fit most of the off-terrain points to the model, which leads to the wrong solution. Therefore, a fitting procedure must be differently applied to the data depending on its labelling attributes.

In the current study, the "terrain polarity" is adopted to measure the "goodness of fit" between a plane terrain model and the data given. As outlined in the previous section, the "terrain polarity" is based on the particular labels (i.e., "ons", "buffs", and "offs"), which are differently generated for each model. Those labels represent an intra-relation of on-terrain labels ("ons") or off-terrain labels ("offs"), while the "buffs" label means a labelling transition from on-terrain to off-terrain and vice versa. As the labels are used as particular observations for the model selection, several advantages can be taken to measure the "goodness of fit". On the one hand, inliers ("ons" and "buffs") and outliers

("offs") can be pre-classified so that the "goodness of fit" for a model is measured depending on the label attributes. On the other hand, more diverse "goodness of fit" measurements can be achieved compared to a monotonic residual measurement, and in particular, a contextual relation of on-terrain and off-terrain points is considered.

The minimum description length (MDL) criterion by Rissanen (1984) provides a generic method for comparing the optimality of different models fitted to particular observations (Cham & Maitre, 1999). In UTF, this MDL criterion is employed in order to determine an optimized model, T^*, amongst the tetrahedron model candidates, $\{T_j\}$, generated for ϕ. As a set of labels, Γ_j, is generated for T_j by a height threshold, δ_h, *a posteriori* probability, $P\left(T_j \mid \Gamma_j, \delta_h\right)$, representing our state of knowledge about the truth of the model hypothesis in the light of Γ_j, is described in the Bayesian estimate as follows:

$$P\left(T_j \mid \Gamma_j, \delta_h\right) \propto P\left(\Gamma_j \mid T_j, \delta_h\right) P\left(T_j\right) \tag{3-16}$$

where Γ_j is a set of labelling observations measured for all the three lateral facets of T_j given, which is obtained by Eq. 3-13 and Eq. 3-14. In Eq. 3-16, $P\left(\Gamma_j \mid T_j, \delta_h\right)$ is the conditional probability that the label, R_j, would be experienced, given that T_j is the correct model hypothesis and the prior probability, $P\left(T_j\right)$, is an initial degree of belief that T_j is the correct plane terrain model. Thus, as the entire potential model hypotheses are estimated by Eq. 3-16, the Bayesian criterion selects T^* as the best model which maximizes *a posteriori* probability, which is called the MAP (Maximum A Posteriori) method. This model selection is given by

$$T^* = \arg\max_{\forall\{T_j\}} P\left(\Gamma_j \mid T_j, \delta_h\right) P\left(T_j\right) \tag{3-17}$$

In the principle of the MDL, the optimal model is the one which needs the smallest length of code in a given encoding language for describing the input data. The MDL has close relationships to the MAP of Eq. 3-17. For example, if $P(T_j)$ is related to the description length for explaining the model, and $P(\Gamma_j \mid T_j, \delta_h)$ related to the description error encoded by Γ_j, then MDL is equivalent to MAP. However, MDL is a more natural and intuitive method when the *prior* probability, $P(T_j)$, is not well defined. In MDL framework, when we take the negative logarithm based 2 on both sides of Eq. 3-17, maximizing the *a posteriori* probability can be converted into minimizing the total coding length of describing T_j by Γ_j. Thus, the smallest length, L^*, for describing an optimal model of T^* is

$$L^*(T^*) = \min_{T_j \in \{T_j\}} \left[-\log_2 P(\Gamma_j \mid T_j, \delta_h) + L(T_j) \right] \tag{3-18}$$

In Eq. 3-18, the first term is a description length to encode the "goodness of fit" between the model T_j and its observations Γ_j, which will be described as a degree of terrain polarity later, and the last term, $L(T_j)$, specifies the description length of the parameters of the tetrahedron model, T_j, as its length increases, when the model complexity gets larger. Thus, the MDL optimality in Eq. 3-18 can be achieved when the terrain polarity as being a plane terrain surface is augmented most strongly and the model T_j used is the simplest one of the candidates, $\{T_j\}$.

Li (1993) suggested that the description length of the entire observations, Γ_j, can be efficiently encoded in the MDL framework, when given the model T_j is divided into two parts, the inlier model part and the outlier part, i.e., $T_j = [T_j^{in} + T_j^{out}]$. Here, T_j^{in} is the inlier model fitted to the observations of "ons" and "buffs" and T_j^{out} is the outlier part fitted to "offs" ones. Thus, given Γ_j, the total description length of T_j is described as follows:

$$L\left(T_j\right) = -\log_2 P\left(\Gamma_j^{ons} \mid T_j, \delta_h\right) - \log_2 P\left(\Gamma_j^{buffs} \mid T_j, \delta_h\right) + L\left(T_j^{in}\right) + L\left(T_j^{out}\right) \tag{3-19}$$

If $\Gamma_j^{ons} = \left\{\gamma_k^{ons}\right\}_{k=1}^{N}$ and $\Gamma_j^{buffs} = \left\{\gamma_k^{buffs}\right\}_{k=1}^{M}$, the conditional probability of $P\left(\Gamma_j^{ons} \mid T_j, \delta_h\right)$ and $P\left(\Gamma_j^{buffs} \mid T_j, \delta_h\right)$ are the product of the probabilities for the individual measurements:

$$\begin{aligned} P\left(\Gamma_j^{ons} \mid T_j, \delta_h\right) &= \prod_{k=1}^{N} P\left(r_k^{ons} \in \Gamma_j^{ons} \mid T_j, \delta_h\right) \\ P\left(\Gamma_j^{buffs} \mid T_j, \delta_h\right) &= \prod_{k=1}^{M} P\left(r_k^{buffs} \in \Gamma_j^{buffs} \mid T_j, \delta_h\right) \end{aligned} \tag{3-20}$$

Then, the description length of Eq. 3-20 becomes

$$\begin{aligned} L\left(T_j\right) = &\sum_{\forall \gamma_k \in \Gamma_j^{ons}} -\log_2 P\left(\gamma_k^{ons} \mid T_j, \delta_h\right) + \sum_{\forall \gamma_k \in \Gamma_j^{buffs}} -\log_2 P\left(\gamma_k^{buffs} \mid T_j, \delta_h\right) \\ &+ L\left(T_j^{out}\right) + L\left(T_j^{in}\right) \end{aligned} \tag{3-21}$$

where γ_k is a labelling observation generated for T_j which is labelled as either γ_i^{ons} or γ_i^{buffs} by Eq. 3-13 and Eq. 3-14. On the right side of Eq. 3-21, the first term and second terms indicate lengths of a degree of the "smoothness" and "discontinuity" of the terrain polarity respectively, which are differently measured depending on γ_k. As presented in the previous section, the terrain polarity of γ_k is measured by an angle difference, $\Delta\theta_k$, from T_j by Eq. 3-15. In fact, the model T_j hypothesizes the terrain slope of underlying area by the characteristic slopes of the lateral facets of T_j. If T_j accurately

approximates underlying terrain surface, $\Delta\theta_k$ measured for γ_k^{ons} (i.e., the "smoothness terrain polarity") shall follow the tendency of underlying terrain slope, while $\Delta\theta_k$ measured for γ_k^{offs} (i.e., the "discontinuity terrain polarity") show an obvious discontinuity. Thus, the conditional probability for the model T_j and an observation γ_k is given by

$$P(\gamma_k \mid T_j, \delta_h) = \begin{cases} \dfrac{1}{1+e^{+\alpha(\Delta\theta_k-\beta)}} & if\ \gamma_k = \{ons\} \\ \dfrac{1}{1+e^{-\alpha(\Delta\theta_k-\beta)}} & if\ \gamma_k = \{buffs\} \end{cases} \tag{3-22}$$

where α and β are the parameters for the sigmoidal function which generates a normalized probability density function; its minimum and maximum probability is restricted up to 0 and 1 respectively.

The conditional probability of Eq. 3-22 is maximized either when $\Delta\theta_k$ of "ons" label describing an intra-relationship of on-terrain points is measured close to 0° or when $\Delta\theta_k$ of "buffs" label, representing an inter-relationship between on- and off-terrain points, is measured close to 90°. In this case, the description length of Eq. 3-21 gets shortened. In Eq. 3-22, α is the parameter to determine the shape of the sigmoidal function, where a larger value of α makes the transition from the maximum to minimum of the terrain polarity shorten (see figure 3-17(a)). The transition of the terrain polarity in the sigmoidal function occurs around β (see figure 3-17(b)). Since β represents a slope change between on-terrain and off-terrain points, the slope value of 45° is usually selected for β.

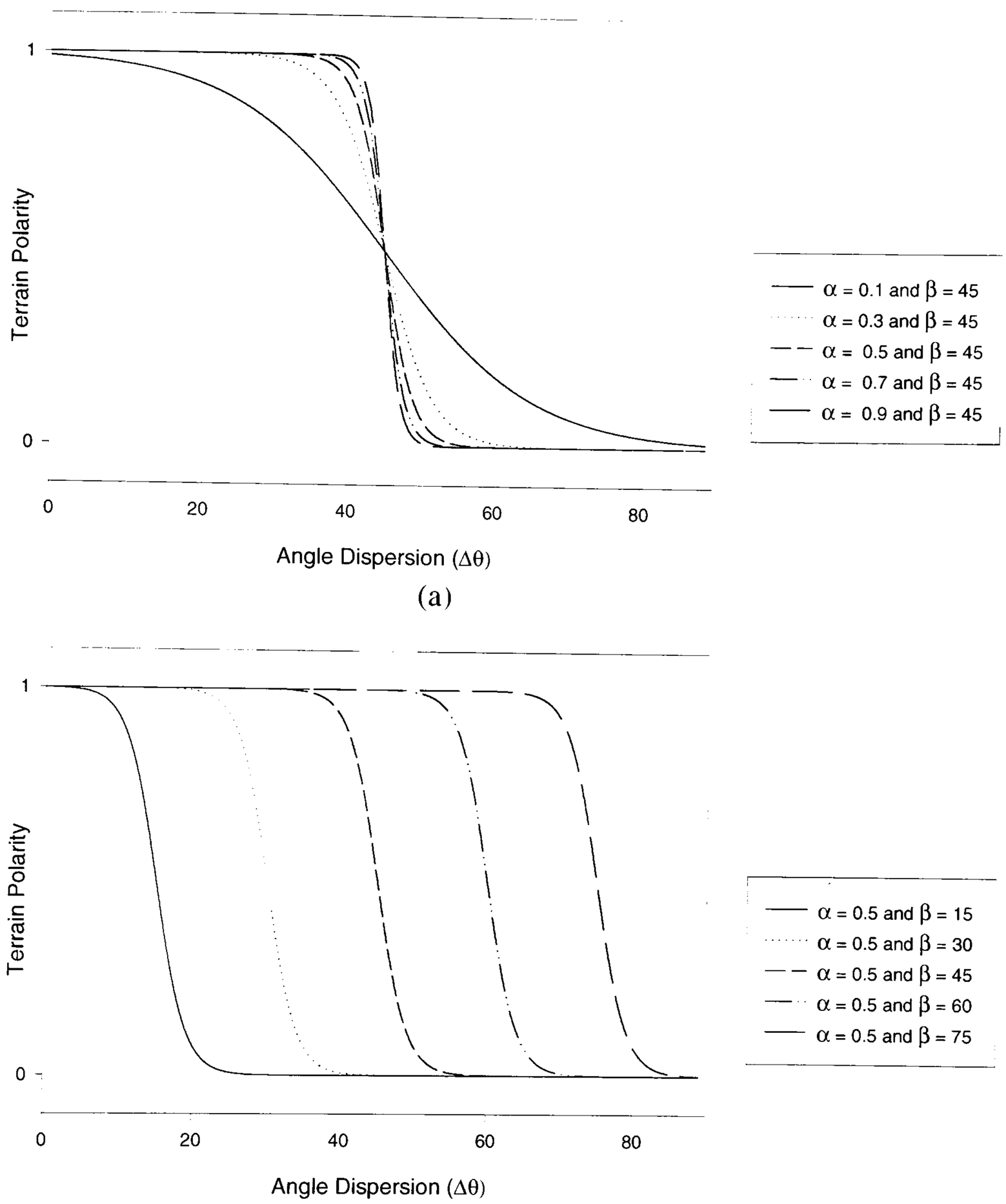


Figure 3-17. Terrain polarity depending on α and β; (a) terrain polarity varying with α, and (b) terrain polarity varying with β

The last two terms in Eq. 3-21 are the description lengths for T_j^{out} and T_j^{in} respectively. $L(T_j^{out})$ is the description length of the number of outliers, that is $L(T_j^{out}) = -\log_2 N_{offs}$, which implies that the MDL criterion prefers a model which populates more off-terrain points when the strengths of the terrain polarity are comparable across model candidates.

Likewise, $L(T_j^{in})$ is the description length of a tetrahedron model used. Since the entire model candidates $\{T_j\}$ share the same base triangle, the only difference that can be characterized for an individual model is the size of volume of T_j, which is proportional to its height H_j. Thus, the description length to encode T_j^{in} is generated by $L(T_j^{in}) = \log_2 H_j$, which means that the MDL criterion favours that the terrain surface is reconstructed smoothly, rather than abruptly. Thus, the model having a smaller volume is preferred in Eq. 3-21. Therefore, the best model, T^*, leading to the minimum description length of Eq. 3-17, is now described by

$$T^* = \arg\min_{T_j \in \{T_j\}} \left[\sum_{\forall \gamma_k \in \Gamma_j^{ons}} -\log_2 \frac{1}{1+e^{+\alpha(\Delta\theta_k - \beta)}} + \sum_{\forall \gamma_k \in \Gamma_j^{buffs}} -\log_2 \frac{1}{1+e^{+\alpha(\Delta\theta_k - \beta)}} - \log_2 N_{offs} + \log_2 H_j \right] \tag{3-23}$$

As the best model, T^*, is selected, on-terrain points involved in constructing T^* are stored in the "on-terrain point stack". This process continues until UTF is performed over all models stored in the "model stack" of figure 3-10. Then, if new on-terrain points are found from the "on-terrain point stack", a set of PTM is generated with those points by the Delaunay Triangulation and stored in the "model stack". The UTF process repeats over the terrain models. However, if there is no on-terrain point in the "on-terrain point stack" after UTF process runs over the entire model stack, the RTF filtering process is terminated, and the DTM is generated with the on-terrain points generated at the previous UTF process.

3.4.4 Post-processing by contextual reasoning

The Type I and Type II errors may be included in the DTM generated by the RTF filter. This section describes a post-processing algorithm, namely "contextual reasoning" process to reclassify these erroneous labels by investigating their contextual relation with surrounding points.

3.4.4.1 TIN colouring

A TIN colouring algorithm was implemented in order to support post-processing. The algorithm aims to segment an irregularly spaced LIDAR data set into a number of homogeneous regions, where only points with the same labels are grouped and a unique region id is assigned to those points. This algorithm was developed based upon Lumia et al.'s work (1983) with some modifications to deal with irregularly spaced dataset.

In the first step, a TIN is constructed with on-terrain and off-terrain points, which are generated through the recursive terrain fragmentation process. When a LIDAR point, s_i, is selected from TIN, its label information, f_i, is extracted. Then, m number of neighbouring points, $N_i = \{n_j\}_{j=1}^{m}$, are obtained, where n_j is connected to s_i and n_j's label, f_j, is the same as f_i. Finally, a region id is assigned to s_i by following two rules; i) if no neighbouring point is marked with a region id, a new region id is created and assigned to s_i, and ii) if one or more neighbouring points have their own region ids, their minimum region id is assigned to s_i. This process is applied to all points of TIN.

At the previous step, the colouring algorithm visits each point of the TIN only once during the process in order to assign a region id. Due to this, the connected points with the same label may have different region ids. To cope with this problem, the second step is to recursively visit each point of the TIN until all the connected points with the same label has a unique region id. To this end, a LIDAR point, s_i, is selected from the TIN and its neighbouring points, N_i, as extracted in the same way as the first step. If N_i's region ids are different from those of s_i, a minimum region id of N_i is assigned to s_i. This second colouring process recursively continues until all the connected points with the same label have a unique region id. The final step is to reassign each region id so that there is no id gap between the region ids.

3.4.4.2 Region-based post-processing

Figure 3-18 illustrates a situation which generates a Type II error during the recursive terrain fragmentation process. This error may be caused by two cases; i) when a corner point created for generating an initial terrain model is located over a real off-terrain feature, or ii) when a real off-terrain point is misclassified as on-terrain point during the

upward fragmentation process. In both cases, the upward fragmentation may proceed, to some extent, over underlying off-terrain features (see figure 3-18 (a)). However, it is soon terminated around the off-terrain features' boundary, since a slope of the local terrain model, ϕ_j, is too steep to capture any on-terrain point by δ_h (see figure 3-18 (b)). Thus, the on-terrain points misclassified inside an off-terrain feature can be isolated by off-terrain points located around the boundary of off-terrain feature. To remove these errors, region-based post-processing was implemented.

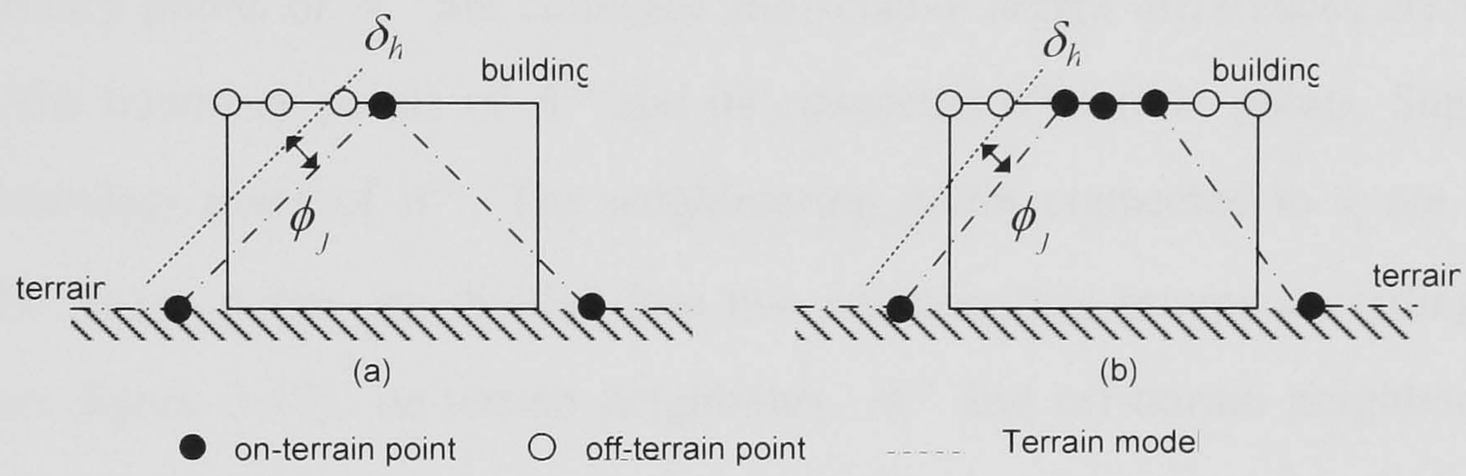


Figure 3-18. Type II error generated around building edges.

The TIN colouring algorithm results in a number of homogeneous regions, which have the same label and unique region id. Suppose that an on-terrain region, A^{on}, with member points labelled as on-terrain point, is selected. It is assumed that natural terrain is always surrounded by off-terrain features and vice versa. Based on this, a A^{on} is surrounded by a number of off-terrain regions, $\{A^{off}\}$. The region-based post-processing algorithm aims to verify A^{on} by reasoning a contextual relation; there must exist an obvious height discontinuity between A^{on} and $\{A^{off}\}$. By that, the process converts an A^{on} into A^{off}, if a height discontinuity between A^{on} and $\{A^{off}\}$ is less than a certain threshold.

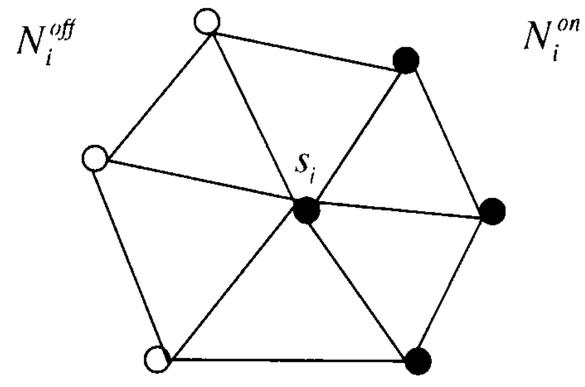


Figure 3-19. Contextual relations of neighbouring labels

The boundary points of A^{on} are collected and relative height differences are measured between the boundary points of A^{on} and its connected off-terrain points. Suppose that s_i is a boundary point of A^{on}. The neighbouring points connected to s_i are extracted from TIN. These points are divided into two neighbouring groups according to their labels (see figure 3-19); on-terrain neighbours, N_i^{on} and off-terrain neighbours, N_i^{off}. Then, a mean height value of on-terrain neighbours, $\mu_{N_i^{on}}$, is measured from N_i^{on}. An absolute maximum height difference between $\mu_{N_i^{on}}$ and heights of N_i^{off} is used as a criterion to assign a penalty score for the contextual reasoning of s_i as follows:

$$K(s_i)\begin{cases} 1 \quad if \ \max \arg_{n_i \in N_i^{off}} \left| z(n_i) - \mu_{N_i^{on}} \right| < \delta_h \\ 0 \quad if \ \max \arg_{n_i \in N_i^{off}} \left| z(n_i) - \mu_{N_i^{on}} \right| \geq \delta_h \end{cases} \tag{3-24}$$

where $z(n_i)$ is the height of n_i, which belongs to N_i^{off} and K is a function to assign a binary penalty score for s_i when a threshold δ_h is given; if the absolute height difference between $\mu_{N_i^{on}}$ and $z(n_i)$ is less than δ_h, a contextual relation of s_i with neighbouring off-terrain points is thought to be unreasonable and the highest penalty score of 1 is given to s_i; otherwise, 0 score is assigned. Thus, the total score of contextual reasoning over A^{on}, which can be measured along M boundary points in a normalized form is as follows:

$$K(A^{on}) = \frac{1}{M}\sum_{i=0}^{M} K(s_i); \quad s_i \in A^{on} \tag{3-25}$$

Note that s_i, which have more than three connected on-terrain points, is only considered in Eq. 3-25 for obtaining a more reliable statistical measurement of $\mu_{N_i^{on}}$. Finally, we can determine whether or not an on-terrain region A^{on} is properly isolated from off-terrain features in comparison with the total contextual reasoning score, $K(A^{on})$, with a ratio threshold, δ_{PC}. The δ_{PC} is selected as an allowable value of $K(A^{on})$ in the light of the regional contextual reasoning. That is, if $K(A^{on})$ is larger than δ_{PC}, A^{on} is converted as A^{off} so the member points of A^{on} are reclassified as off-terrain points as follows:

$$A^{on} \Rightarrow A^{off} \quad if\ K(A^{on}) < \delta_{PC} \tag{3-26}$$

3.4.4.3 Point-based post-processing

Another post-processing algorithm was implemented, however at this time it aims to remove the Type I errors based upon a point-by-point reclassification. During the UTF, Type I errors may be generated over certain areas, which have a very steep terrain surface, such as a cliff or an artificially built-up terrain wall. Similar to the case generating the Type II errors around building edges discussed in the previous section, real on-terrain points are mistakenly recognized as off-terrain points around steep edges of terrain surface (see figure 3-20).

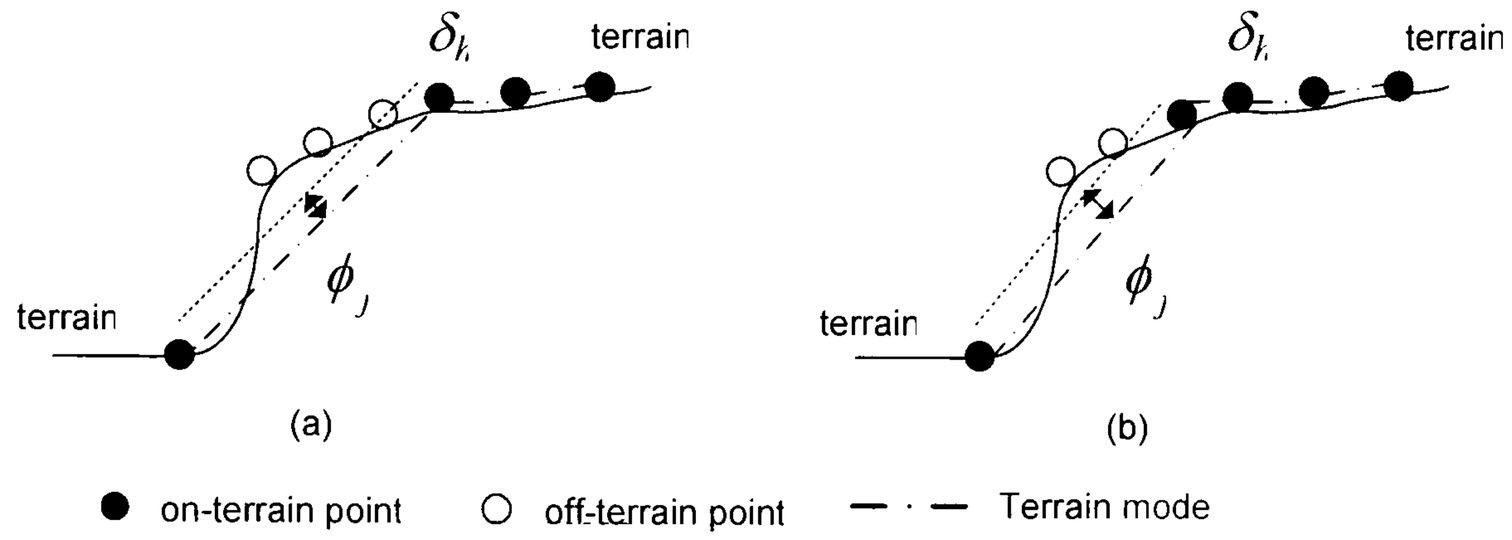


Figure 3-20. Type I error generated around steep terrain edge

In contrast to the region-based post-processing, the aforementioned misclassified points are not isolated well by surrounding on-terrain points, but continuously connected to other sides of the terrain. Thus, a local post-processing algorithm is required. To this end, a point-based post-processing algorithm was implemented. The algorithm aims to reclassify a misclassified off-terrain point as an on-terrain point by investigating the local contextual relation between on-terrain and off-terrain points.

Suppose that an off-terrain point s_i and its connected neighbouring on-terrain points, N_i^{on}, are extracted from a TIN generated by the region-based post-processing. A relative height distance between the height of s_i and the mean height of N_i^{on}. As this relative height is compared with the standard deviation of N_i^{on}'s heights, s_i, is reclassified as a new label, f_i, as follows:

$$f_i = \{on\} \qquad if \left| z(s_i) - \mu_{N_i^{on}} \right| < \sigma\left(z\left(N_i^{on}\right)\right) \tag{3-27}$$

where $z(s_i)$ and $z(N_i^{on})$ are the height of s_i an d N_i^{on} respectively; $\mu_{N_i^{on}}$ is a mean height of N_i^{on}; $\sigma\left(z(N_i^{on})\right)$ is the standard deviation of N_i^{on}'s heights. In Eq. 3-27, an off-terrain point, s_i, is reclassified as on-terrain point if a height difference between $\mu_{N_i^{on}}$ and $z(N_i^{on})$ is less than $\sigma\left(z(N_i^{on})\right)$. This process is iteratively applied to all off-terrain points until no label is changed.

3.5 Experimental results

This section aims to illustrate each processing step of the RTF filter using different LIDAR data. A complex area is chosen from the OEEPE LIDAR DSM (OEEPE, 2000), which is shown in figure 3-21. The characteristics of the data used are described in table 3-1.

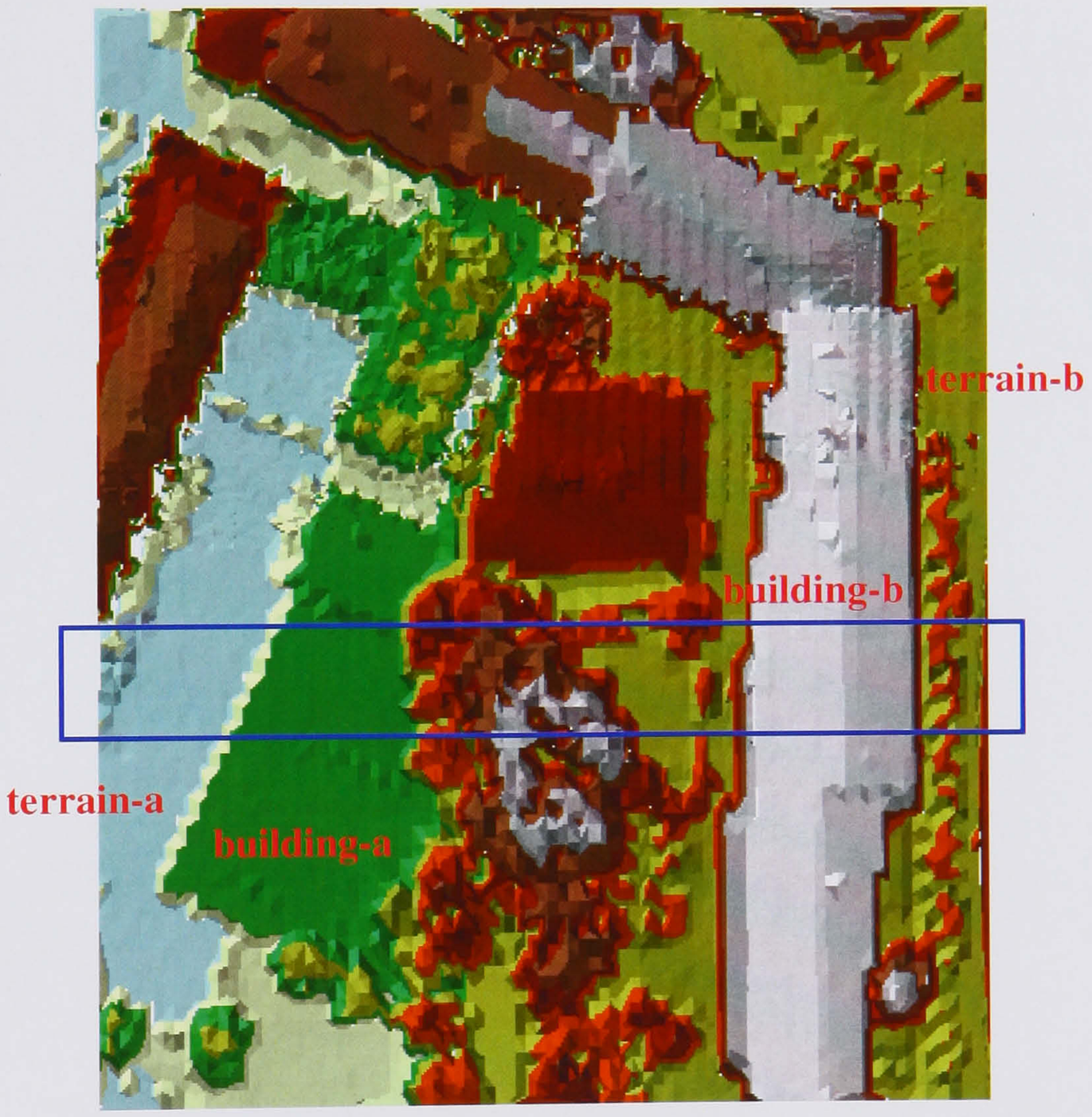


Figure 3-21. OEEPE LIDAR DSM

Table 3-1. Data characteristics of the OEEPE LIDAR DSM

Data characteristics	Value
Total number of LIDAR points	8,967 (points)
Point density	~0.9 (point/m^2)
X range	513723.42 ~ 513813.26 (m)
Y range	5402761.40 ~ 5402871.93 (m)
Z range	298.570 ~ 331.840 (m)

As can be seen in figure 3-21, the test area is mixed up with two different terrain surfaces of terrain-a and terrain-b, with a height difference larger than 10 metres. These terrains are disconnected by a small group of trees and buildings between them. To highlight this circumstance, a height profile covering a blue box region of figure 3-21 is illustrated in figure 3-22. In this region, the terrain-a and terrain-b are clearly differentiated from each other in height and the building-a is directly adjacent to the terrain-b.

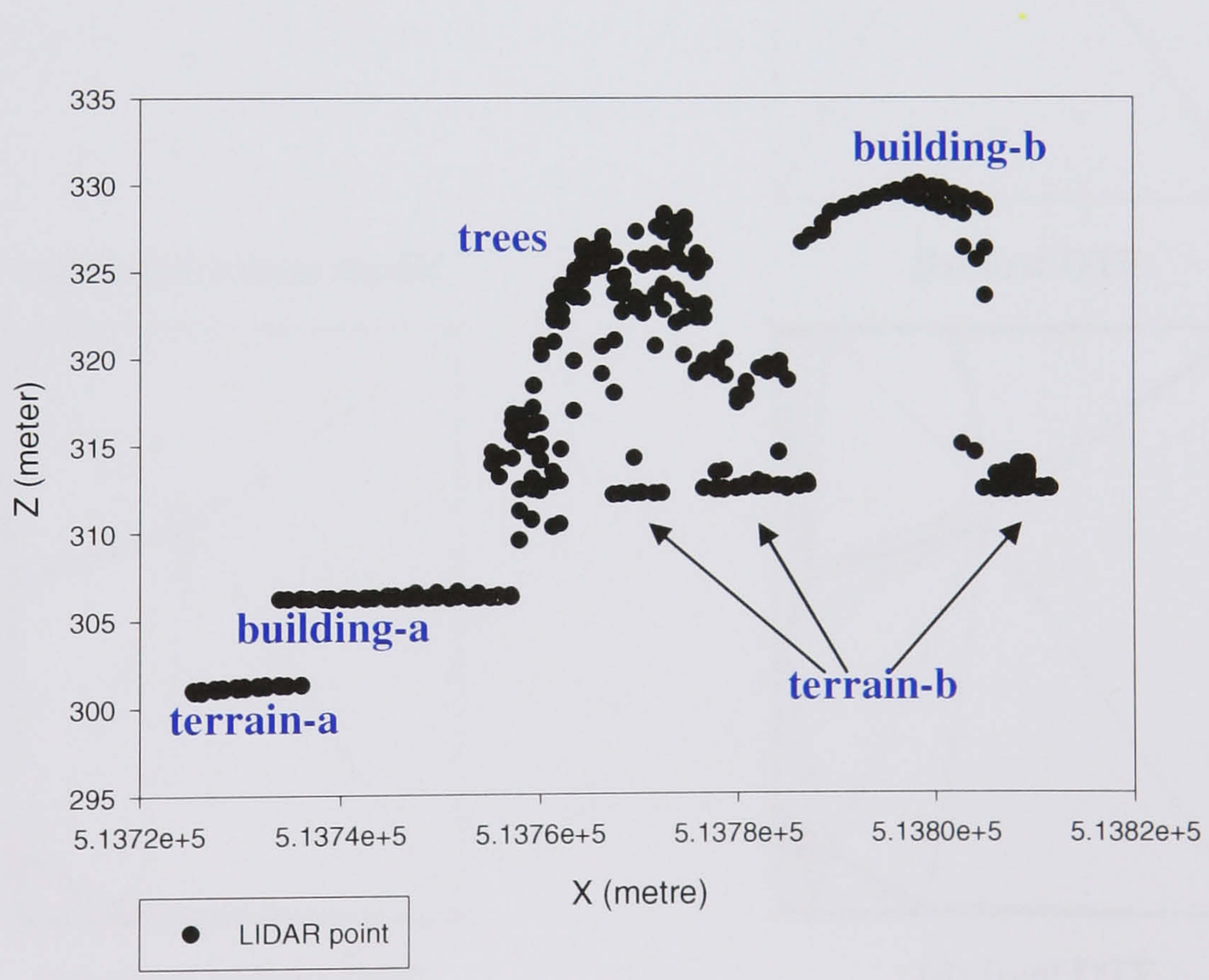
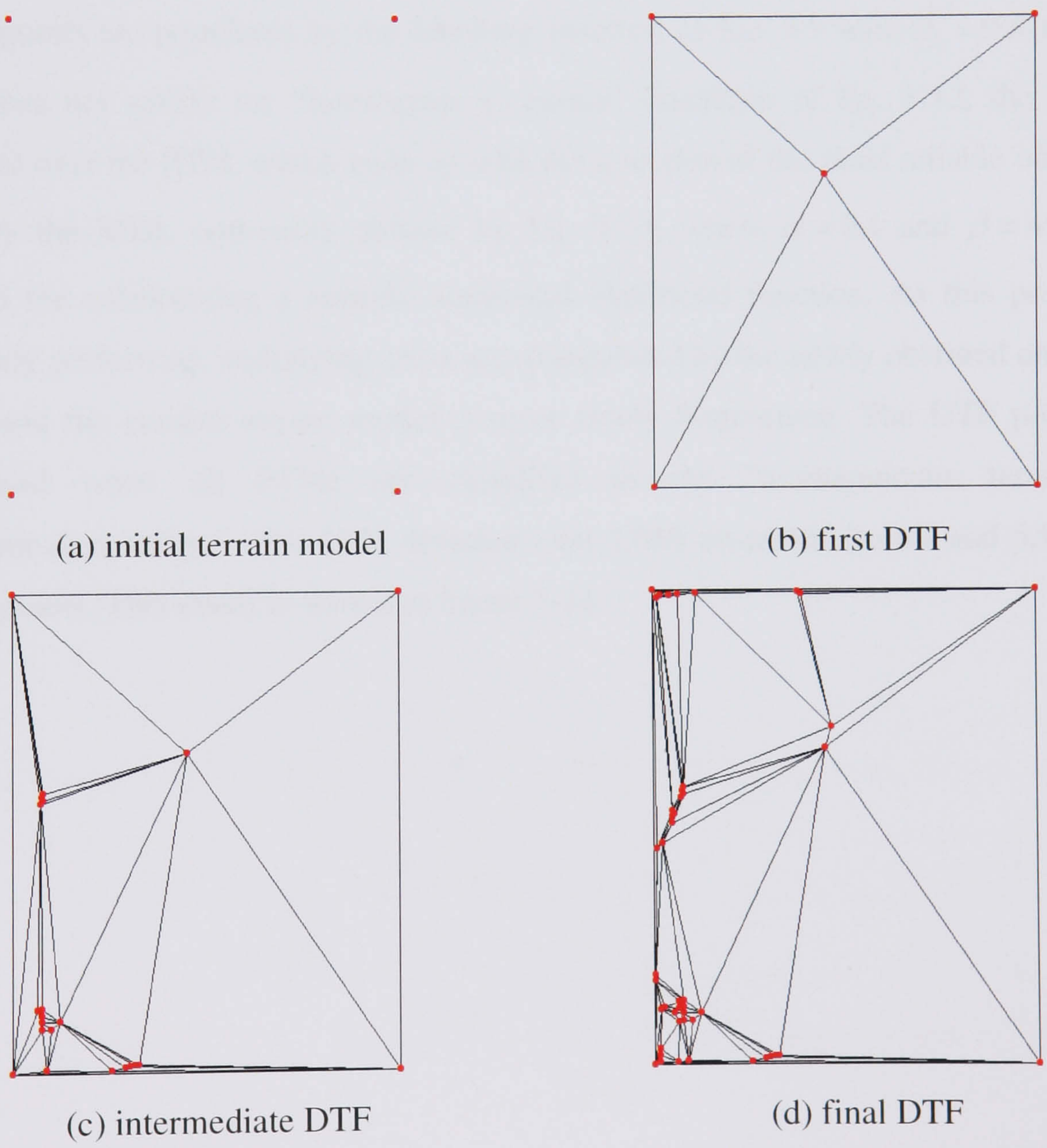


Figure 3-22. Height profile before the RTF filtering

The RTF filter was applied to the LIDAR DSM of figure 3-21 to reconstruct the DTM. The first step is the downward terrain fragmentation (DTF) process. An initial terrain model formed in a rectangle shape is generated by creating four corner points as

discussed in §3.4.1. Those corner points are shown in figure 3-23(a) and listed in table 3-2. Then, a point with the minimum distance below the initial terrain model is selected as an on-terrain point, by which the LIDAR DSM is triangulated (see figure 3-23(b)). As a consequence, the underlying area is fragmented into four triangular planes, which are employed as the plane terrain models (PTMs) for the successive DTF process (see figure 3-23(c)). The process is terminated when all the PTMs are verified as a "positive terrain" defined in Eq. 3-8, and total 47 on-terrain points are detected by the DTF. This result is shown in figure 3-23(d).



(a) initial terrain model

(b) first DTF

(c) intermediate DTF

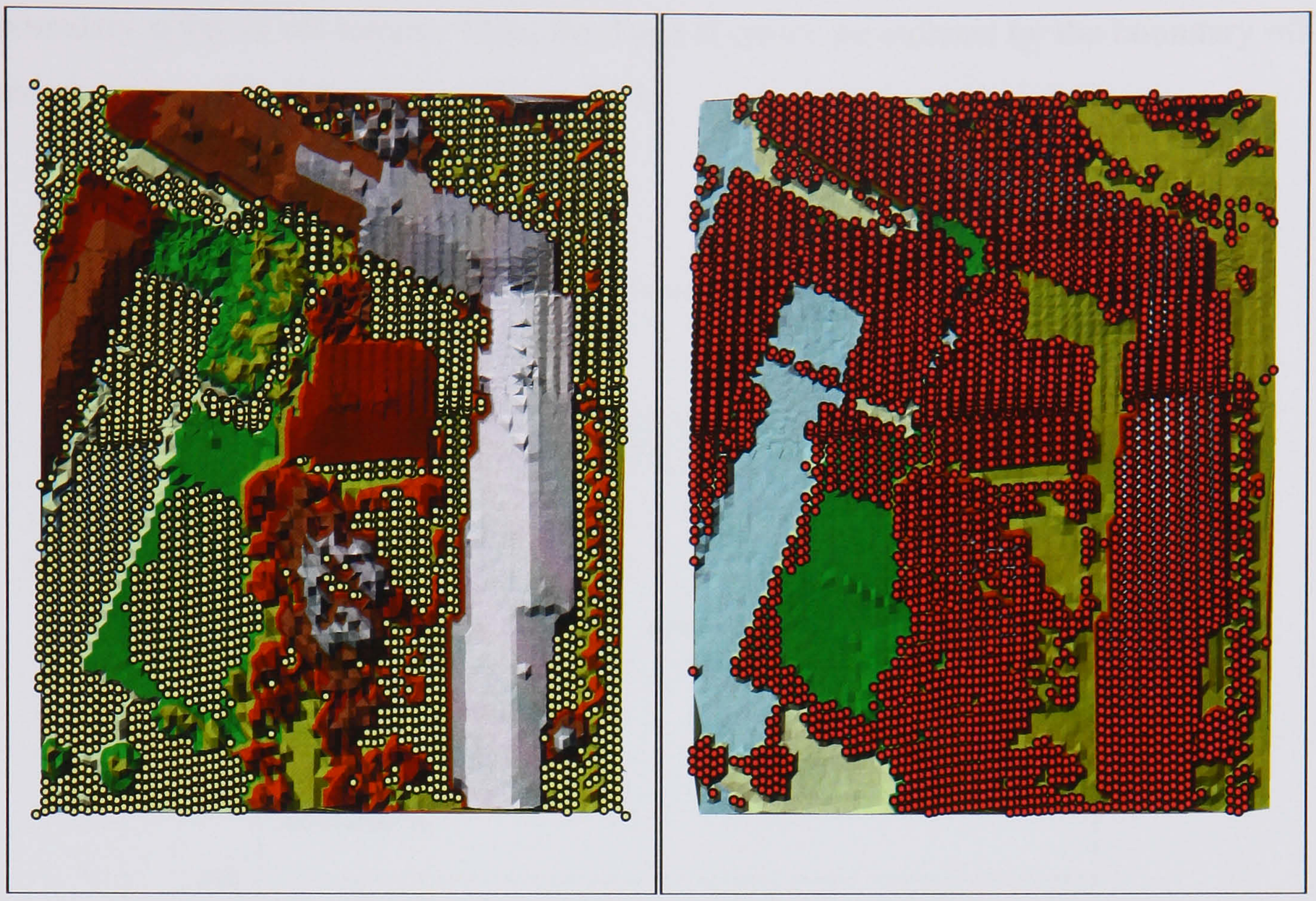
(d) final DTF

Figure 3-23. Downward terrain fragmentation result; red dots represents on-terrain points

Table 3-2. Initial terrain model information

Vertex index	Coordinates (X,Y,Z) in meter
v_1	(513723.420000, 5402761.400000, 303.539978)
v_2	(513723.420000, 5402871.930000, 302.190002)
v_3	(513813.260000, 5402871.930000, 312.869995)
v_4	(513813.260000, 5402761.400000, 312.220001)

The upward terrain fragmentation (UTF) is triggered over the result of the DTF process shown in figure 3-23(d), where each triangle is employed as a PTM. For each PTM, on-terrain points are populated by the labelling function of Eq. 3-9 with $\delta_h = 0.5\,(\mathrm{m})$. If a PTM does not satisfy the "homogeneous terrain" condition of Eq. 3-12, the UTF is triggered over the PTM, which ends up with the selection of the most reliable on-terrain point by the MDL optimality defined by Eq. 3-23, where $\alpha = 0.1$ and $\beta = 45^\circ$ were selected for constructing a specific sigmoidal likelihood function. As this process is iteratively performed, underlying areas are triangulated by the newly obtained on-terrain points and the current terrain model is more finely fragmented. The UTF process is terminated when all PTMs are classified as the "homogeneous terrain" or "discontinuous terrain". The UTF detected total 3,065 on-terrain points and 5,902 off-terrain points. This result is shown in figure 3-24.



(a) on-terrain points (b) off-terrain points

Figure 3-24. Upward terrain fragmentation result

Although the study area contains very different terrain slopes, figure 3-24 shows that the UTF clearly picked up on-terrain points from both terrain-a and terrain-b regions, while buildings, trees and bushes are correctly classified as off-terrain points. However, Type II errors can be found over the rooftop of the building-a, where a number of off-terrain points are mistakenly recognized as on-terrain points (see figure 3-24(a)). This problem is caused by the fact that one side of the building-a is attached to the terrain-b. In this case, some LIDAR points, which are closely located to the terrain-b, can be continuously detected within a height buffer constructed by δ_h from an intermediate terrain model generated. This makes the UTF classify underlying local surface as the "continuous terrain", rather the "discontinuous terrain". In consequence, the UTF process is triggered and on-terrain points are detected over the rooftop of the building-a. In this way, the Type II errors spread over the building-a region to some extent. However, since the other parts of the building-a are clearly disconnected from the surrounding terrain surfaces of terrain-a, the sharp discontinuity of the building edges makes the UTF classify these boundary regions as the "discontinuous terrain". As a result, the UTF is terminated over the building boundary regions, and labels the building

boundary point as off-terrain. Thus, the Type II errors are isolated by the boundary off-terrain points of building-a (see figure 3-25).

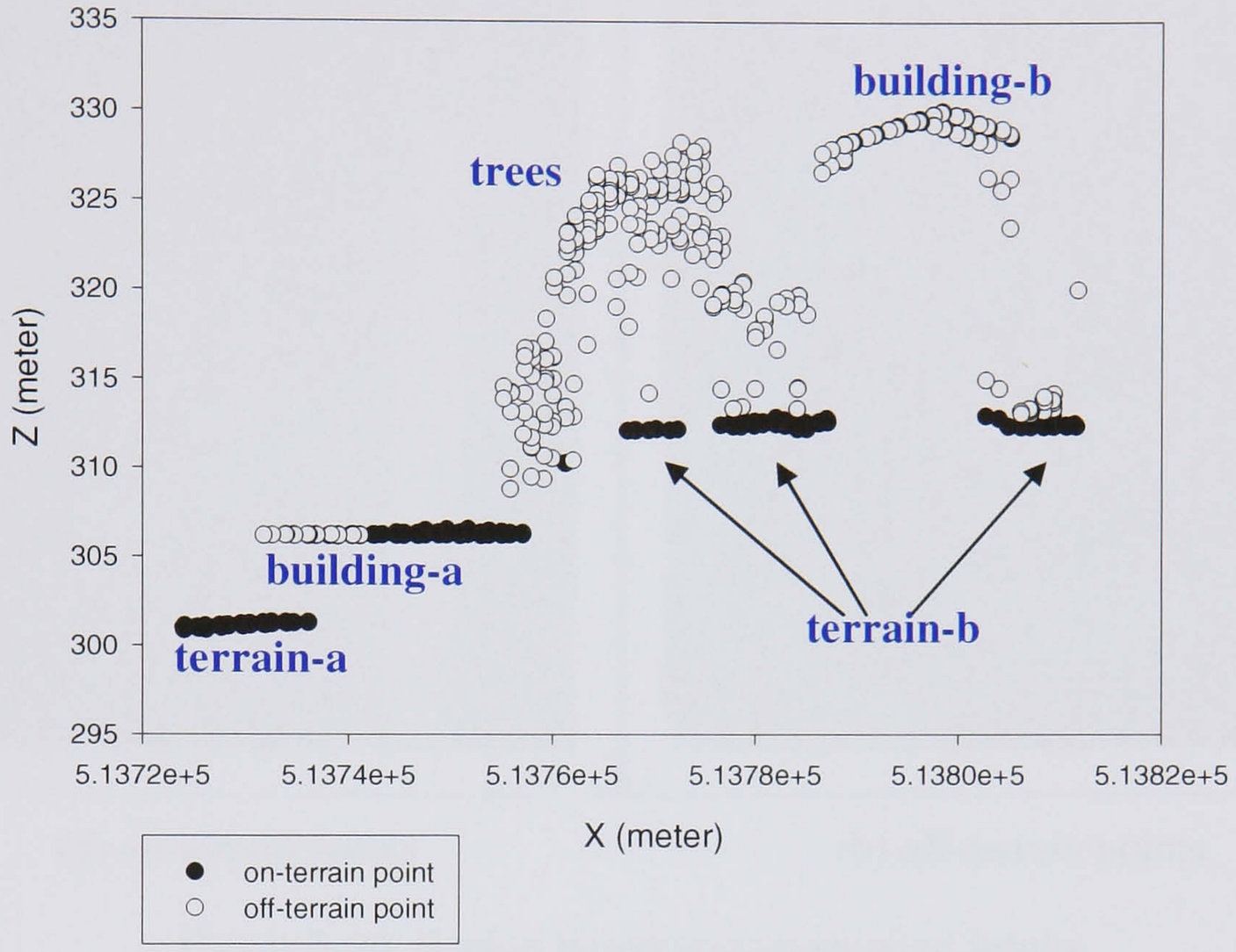
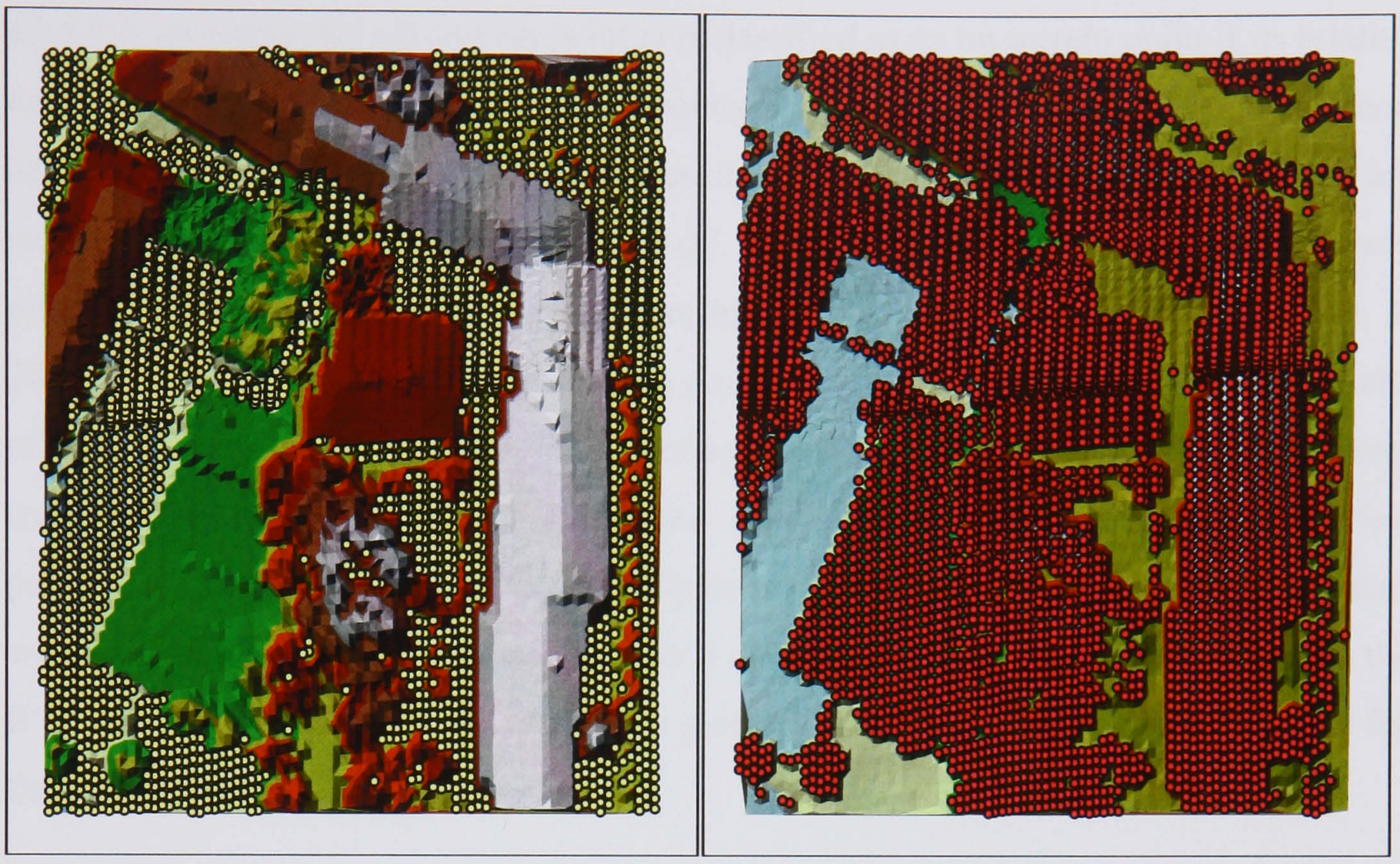


Figure 3-25. Height profile after the upward terrain fragmentation

As discussed in §3.4.4.2, the region-based post-processing algorithm was developed for removing the aforementioned Type II errors. After applying the TIN colouring algorithm, the isolated errors of the building-a are grouped as one region. A contextual reasoning is performed between the boundary points of this region and their off-terrain neighbours by Eq. 3-24. Then, the reasoning result is calculated as a normalized penalty score by Eq. 3-25. As the normalized score is compared with a ratio threshold, $\delta_{pc} = 0.3$, the 295 on-terrain points are now reclassified as off-terrain points. Here, δ_{pc} is pre-determined, assuming 30 % of building boundary points could be attached to the surrounding terrain surface in the study area. Figure 3-26 and 3-27 shows that the region-based post-processing algorithm clearly removes the Type II errors of building-b.



(a) on-terrain points (b) off-terrain points

Figure 3-26. Region-based post-processing results

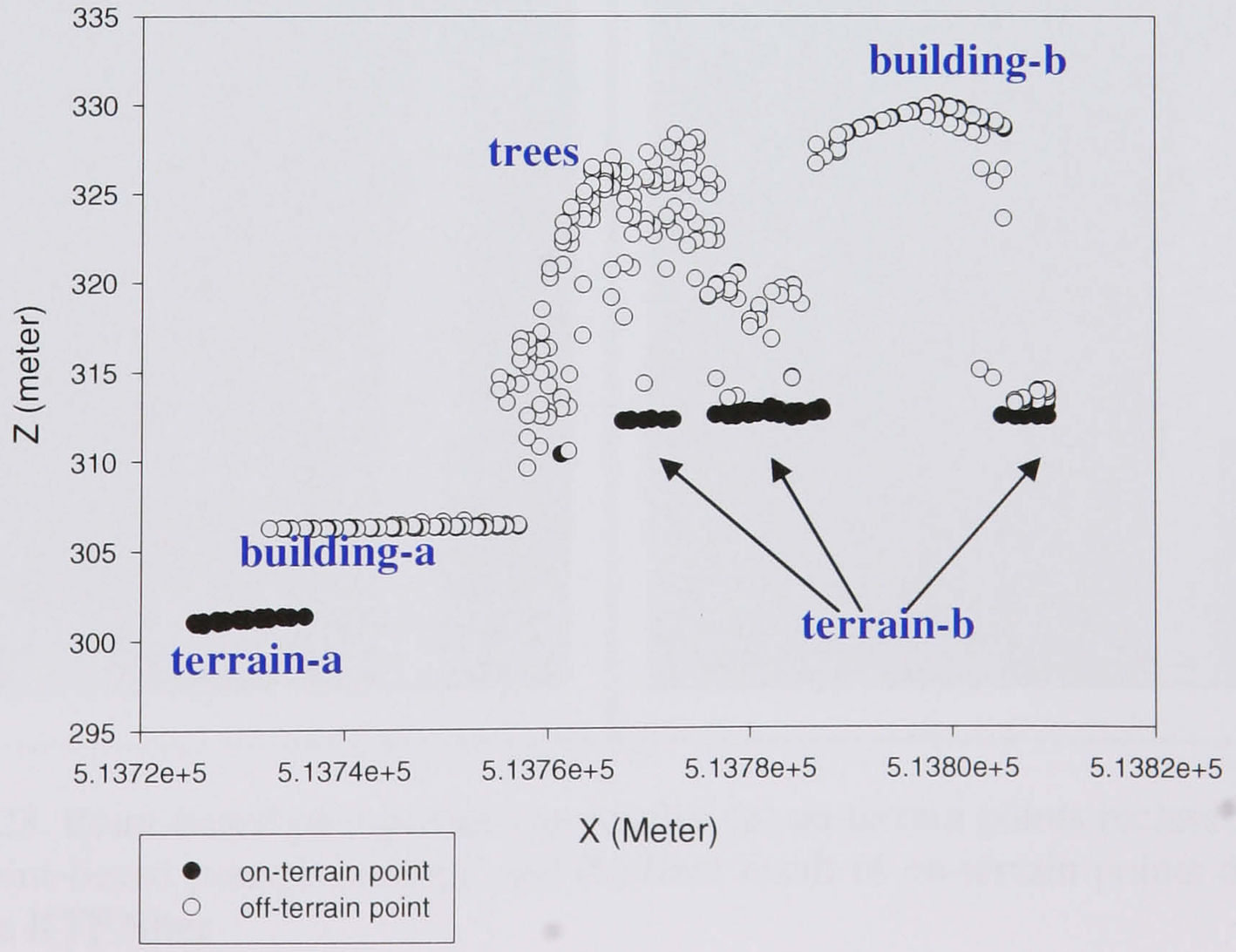


Figure 3-27. Height profile after the region-based post-processing

The last step is to apply the point-based post-processing to the region-based post-processing result shown in figure 3-26 for removing the Type I error. As discussed in §3.4.4.3, an erroneous off-terrain point is reclassified as an on-terrain point if its relative heights with neighbouring on-terrain points fall in the standard deviation of their heights. By the point-based post-processing, a total of 17 off-terrain points are reclassified as on-terrain points by Eq. 3-27. Figure 3-28 (a) shows the result of newly reclassified on-terrain points by the point-based post-processing algorithm. Figure 3-28(b) shows the final result of on-terrain points detected by the RTF filter. The results of each process of the RTF filter is described in table 3-3 and the filtering parameters used are shown in table 3-4. It can be concluded that with relatively very few parameters, the RTF filter can successfully detect on-terrain points from terrain surfaces with very high slope discontinuity, which is common in urban areas. Consequently, the RTF can clearly recognize the off-terrain points with different sizes, which is useful for the purpose of building detection.

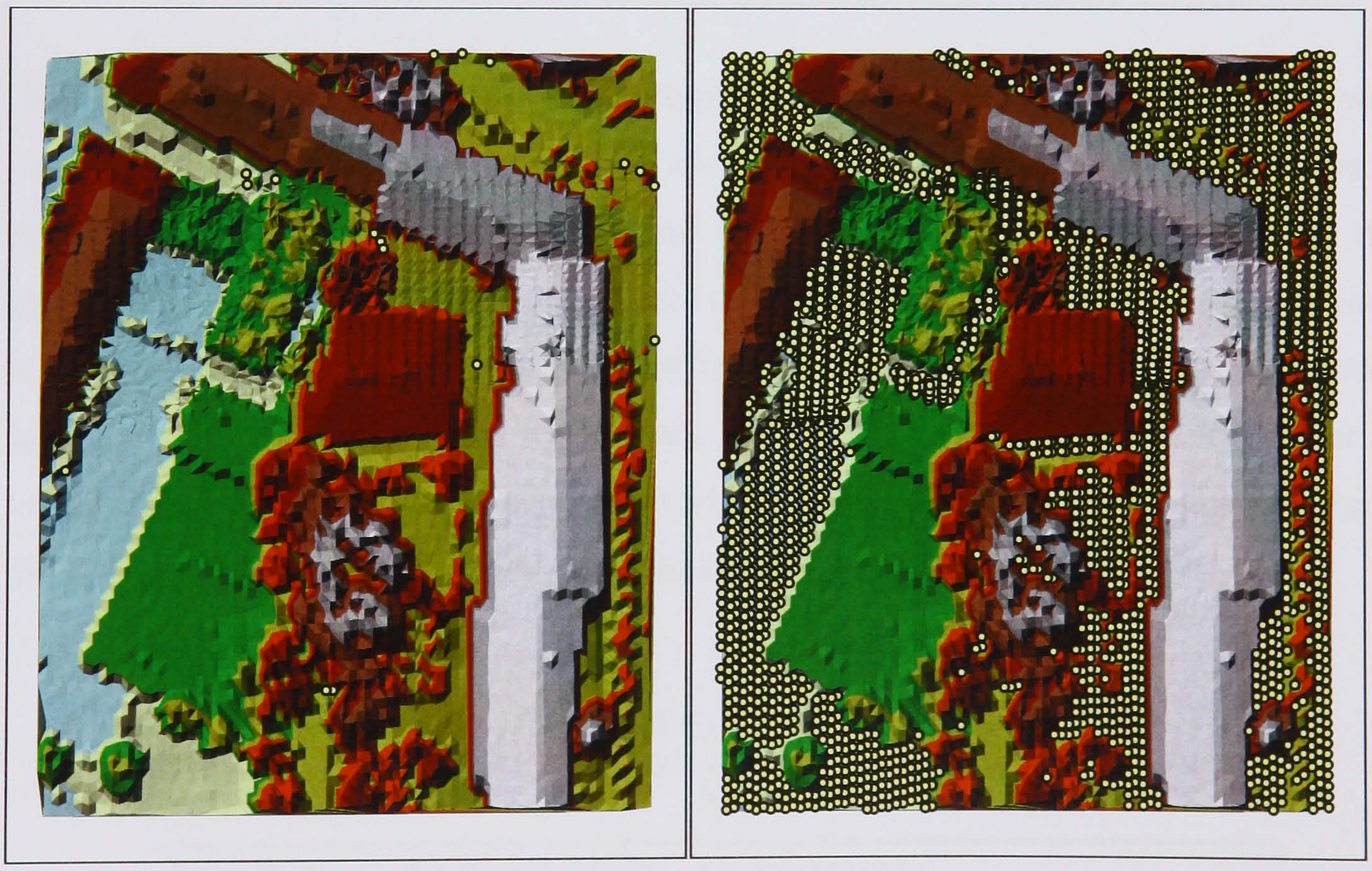


Figure 3-28. Point-based post-processing results; (a) on-terrain points reclassified by the point-based post-processing, and (b) final result of on-terrain points detected by the RTF filter

Table 3-3. RTF filtering result of OEEPE LIDAR DSM

Processing step	Results (points)	
	on-terrain	off-terrain
Downward terrain fragmentation	47	8920
Upward terrain fragmentation	3065	5902
Region-based post-processing	2770	6197
Point-based post-processing	2787	6180
Total		8967

Table 3-4. OEEPE LIDAR DSM

Parameter	Value	Process	Determination
δ_h	1 (meter)	Upward terrain fragmentation	Pre-defined
α	0.1	Upward terrain fragmentation	Pre-defined
β	45 (degree)	Upward terrain fragmentation	Pre-defined
δ_{PC}	0.3	Region-based post-processing	Pre-defined

The RTF filter was tested using a different LIDAR dataset. The LIDAR data was collected over the East London area by the Optech 1020 sensor with one point per 3 metre point spacing. Figure 3-29(a) and 3-30(a) show the LIDAR DSM and the panchromatic imagery of IKONOS satellite of the test area. This area was selected since it contains a "good" mixture of different features such as residential area, flat grass, knolls, forest and hills. These features are suitable to validate how the RTF filter is continuously adaptive to terrain surface variations, especially for gently sloped terrain. The characteristics of the data used are described in table 3-5.

Table 3-5. Data characteristics of the East London LIDAR DSM

Data characteristics	Value
Total number of LIDAR points	16,590 (points)
Point density	~0.1 (point/m^2)
X range	541567.75 ~ 541949.73 (m)
Y range	181332.27 ~ 181740.25 (m)
Z range	-0.0700 ~ 19.3600 (m)

The RTF filter was applied to the East London data. Compared to the OEEPE data, a higher value for δ_h was selected as 1 metre since the point density of the East London data is lower than the OEEPE data, while other parameters were fixed as the same values used in the OEEPE data. The filtering parameters used for the East London data is shown in table 3-6.

Table 3-6. RTF filtering parameters for the East London LIDAR DSM

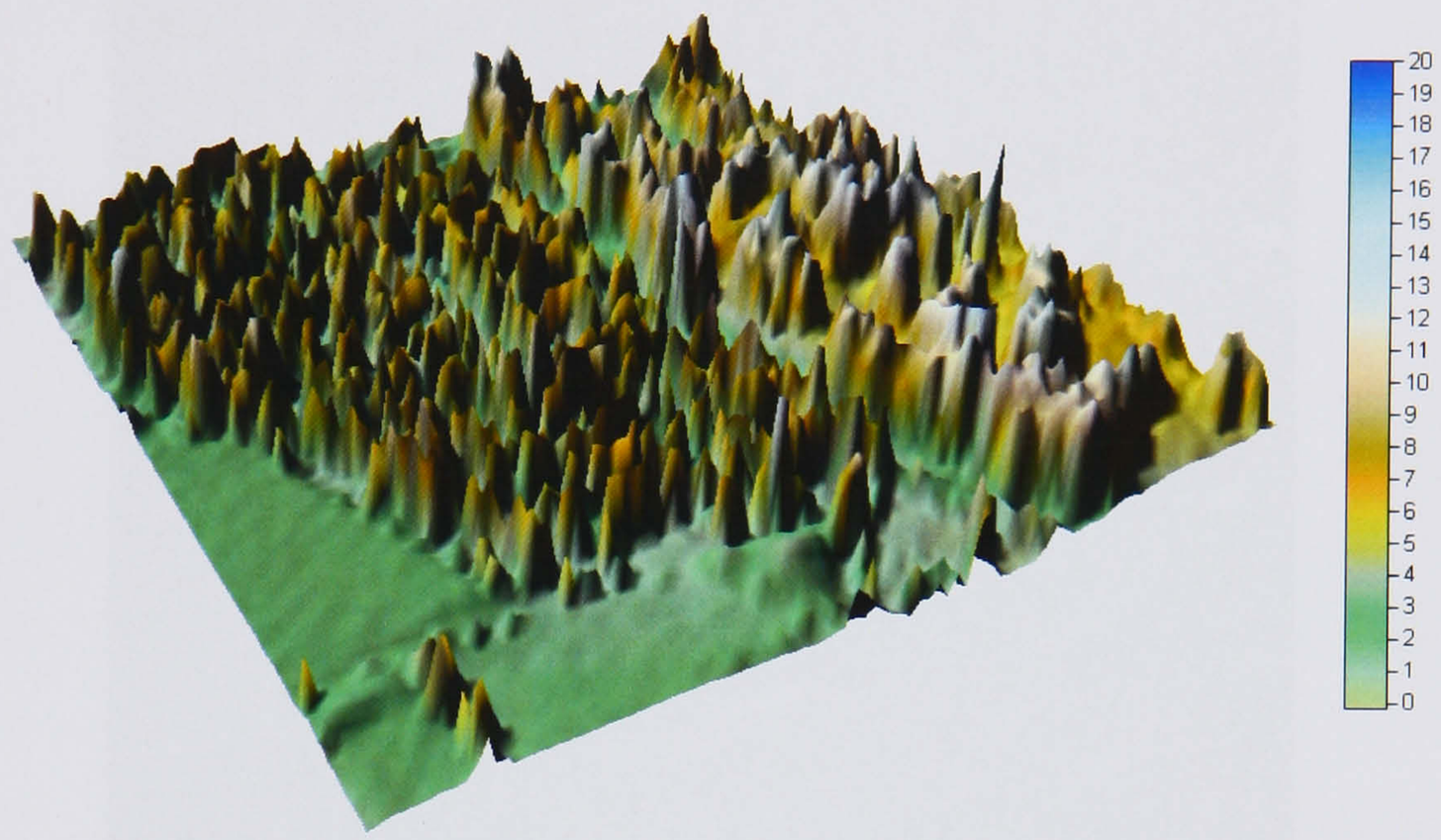
Parameter	Value	Process	Determination
δ_h	1 (meter)	Upward terrain fragmentation	Pre-defined
α	0.1	Upward terrain fragmentation	Pre-defined
β	45 (degree)	Upward terrain fragmentation	Pre-defined
δ_{PC}	0.3	Region-based post-processing	Pre-defined

The results of the RTF filter over the East London data are presented in table 3-7. The DTM reconstructed by the filter is shown in figure 3-29(b) and the off-terrain points extracted are overlaid with the IKONOS panchromatic image in figure 3-30(b). Although the overall terrain is not flat and there are several gentle hills, it can be concluded that the RTF filter clearly extracted off-terrain points from residential houses, trees and bushes, while no point on the hills is labelled as off-terrain (see middle of the bottom in figure 3-30(b)).

Table 3-7. RTF filtering result of the East London LIDAR DSM

Processing step	Results (points)	
	on-terrain	off-terrain
Downward terrain fragmentation	103	16487
Upward terrain fragmentation	10418	6172
Region-based post-processing	10418	6172
Point-based post-processing	10424	6166
Total		16590

(a)



(b)

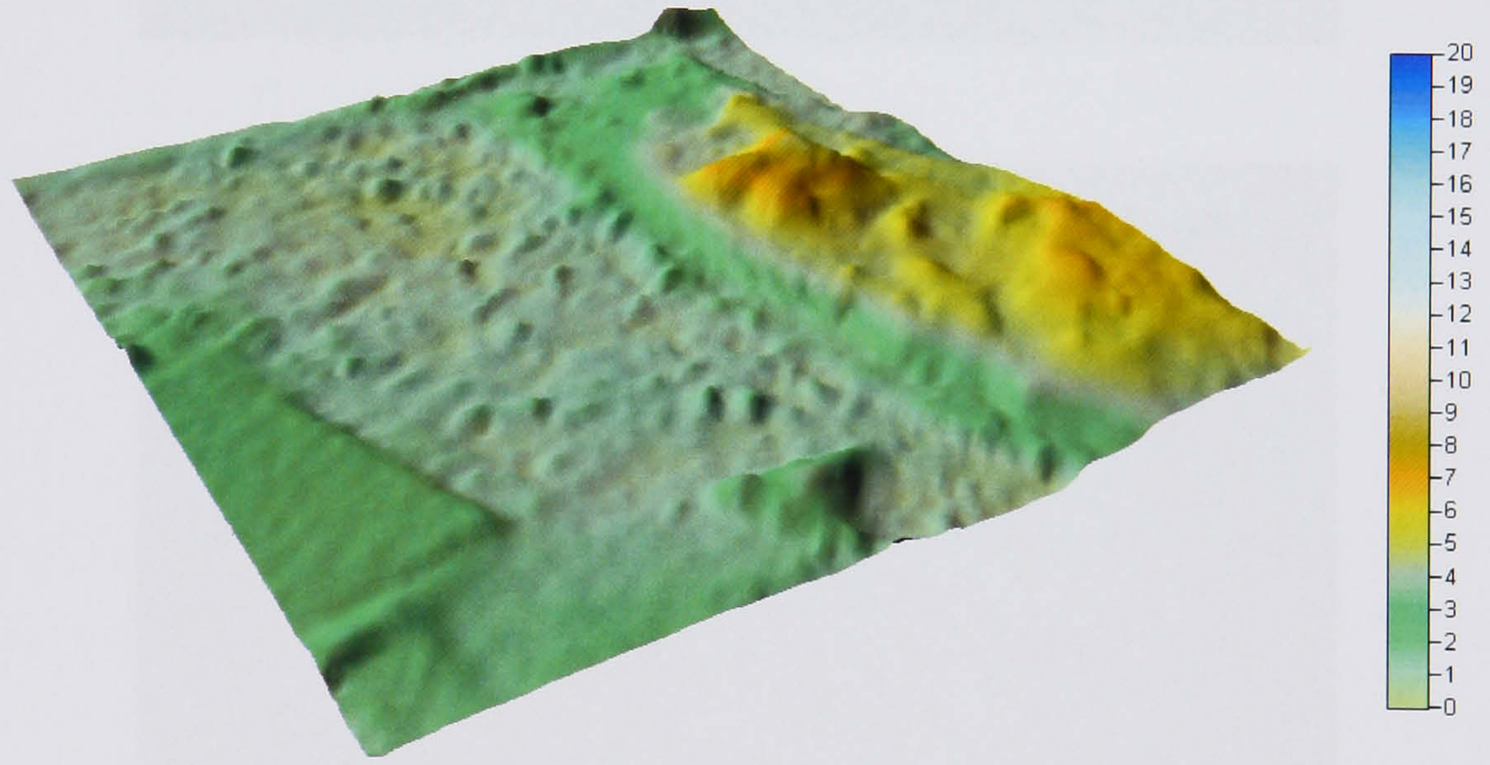


Figure 3-29. DTM reconstruction result of the RTF filter over the East London data; upper image shows the LIDAR DSM, and lower image shows the DTM generated by the filter

(a)

(b)

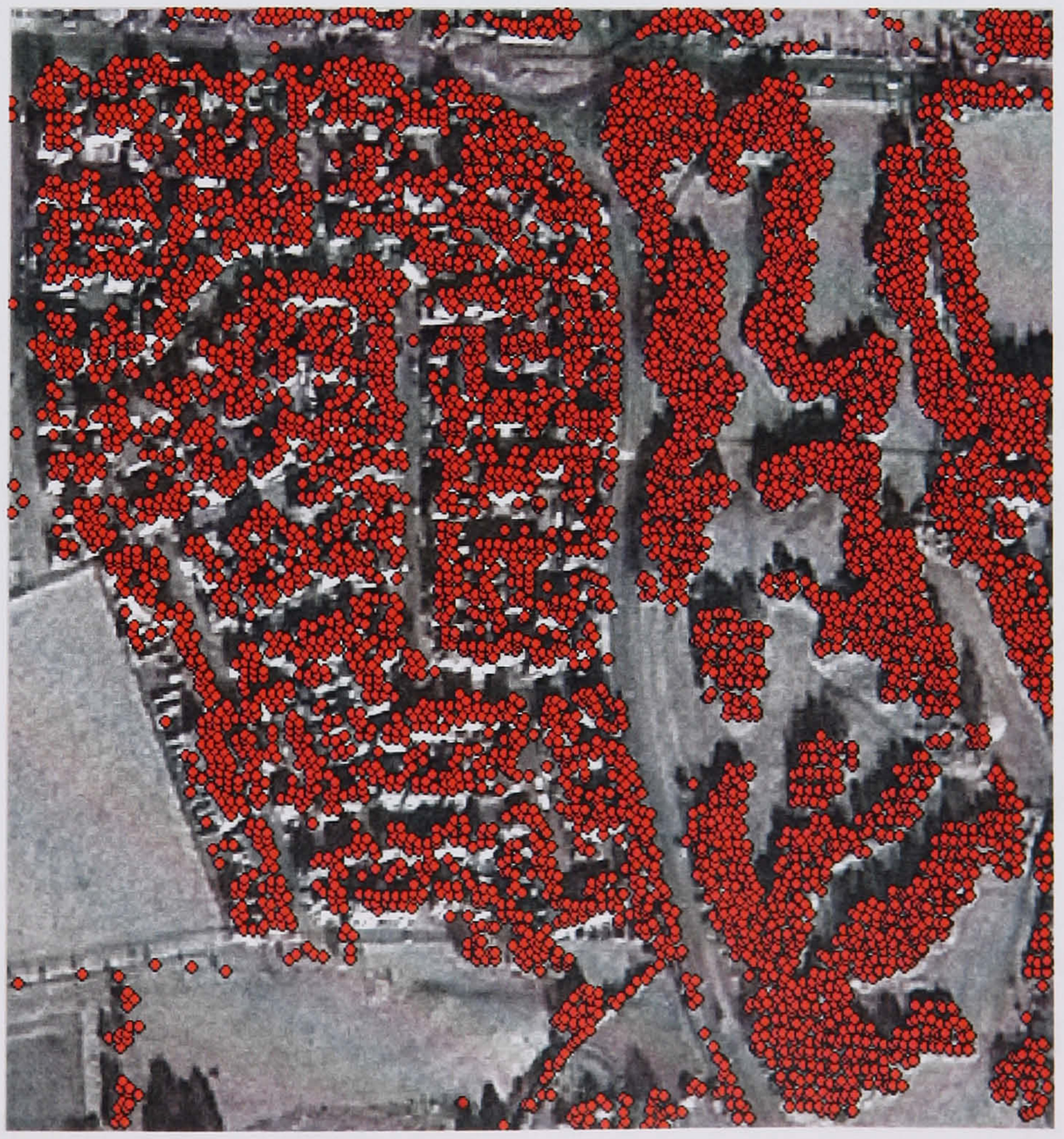


Figure 3-30. RTF filtering result over the East London data; upper image shows the IKONOS panchromatic imagery, and lower image shows the off-terrain points classified by the filter

3.6 Summary

The DTM can be reconstructed by detecting terrain surfaces or removing non-terrain features from LIDAR measurements. This can be achieved by a LIDAR filter, which automatically assigns adequate semantic labels, on-terrain and off-terrain labels, to a cloud of LIDAR points. The major difficulty in developing the LIDAR filter is caused by the fact that there is no "universal" object representation for natural terrain or non-terrain features. This problem makes the filter difficult to apply appropriately in complex scenes with various landforms and object shapes.

In §3.2, existing LIDAR filters were broadly categorized into the "deterministic" filters and the "adaptive" filters, depending on whether the object representations used are determined explicitly or implicitly. The filtering rationales behind both techniques have been discussed with respect to two scale factors, "terrain variation effect" and "object scale effect". In the light of this, some drawbacks of both filtering algorithms have been described; the deterministic filters can work only in a very limited scene, though it is efficient for selecting criteria to differentiate natural terrain from non-terrain features; the adaptive filters can cope with many diverse landforms, but they generally require a large number of parameters and different parameters have to be given for various landforms.

In §3.3 and §3.4, a new technique, called the RTF (Recursive Terrain Fragmentation) filter, was proposed as a further development of this study. The RTF filter takes advantage of both the "deterministic" and "adaptive" filtering techniques, by which natural terrain with various slopes can be reconstructed from irregularly spaced LIDAR measurements with a minimal use of filtering parameters. A few new ideas have been incorporated in developing this technique. A novel concept is associated with the representation of generic terrain surface. A plane terrain surface is employed as an elementary terrain model, and a generic terrain surface is reconstructed by obtaining a set of plane terrain surfaces from a cloud of LIDAR points. The plane terrain model is explicitly defined wherein the "terrain variation effect" is removed. This method is suitable for representing complex terrain surface with various slopes by a single constraint, which is pre-determined as the maximum slope variance expected in a study area.

A "coarse-to-fine" strategy was adopted to help the filter to achieve self-adaption to both the "terrain variation effect" and the "object scale effect". As an initial terrain model comprising entire LIDAR space is more finely fragmented with smaller sub-regions, the "terrain variation effect" over those fragmented regions is more uniformly regularized, and is finally removed at the finest level. As the reconstructed terrain space is eliminated from the entire LIDAR space, non-terrain features are simply obtained regardless of their own scales.

An optimized process was implemented to prevent filtering errors occurred at lower level from propagating to upper level in the "coarse-to-fine" fragmentation. An actual fragmentation decision over a local region is made, as all plausible fragmentation solutions are predicted in advance, and are tested in terms of the "terrain polarity". The best solution is statistically determined by the MDL (Minimum Description Length) method as it approximates to the real terrain surface most closely, with minimum filtering errors.

The RTF filter included some post-processing algorithms to remove remaining filtering errors. It is achieved by a "contextual reasoning" process, which corrects resulting errors in consideration of pre-defined height relations between consecutively connected points with different labels. This method was considered to reduce filtering errors particularly caused by discontinuous terrains, where terrain slopes change sharply.

In §3.5, the RTF filter was applied to two small sub-scenes with various landforms and non-terrain features. The results have shown that the algorithm proposed in this chapter successfully removed various non-terrain features and reconstructed terrain surfaces with slope discontinuities. The main purpose of the filter development is to reconstruct terrain surfaces in a complex urban environment, and thus, to support a building extraction process. The next chapter aims to test the present technique in full scenes with more complex objects in order to accurately assess resulting errors, and to validate how the filtering rationale given by the RTF filter operates in an urban environment.

CHAPTER 4

ACCURACY ASSESSMENT OF AUTOMATIC DTM GENERATION RESULTS

4.1 Introduction

The previous chapter proposed the recursive terrain fragmentation (RTF) filter for an automatic DTM generation and explained in detail how the filter operates using small LIDAR DSMs. In this thesis, the main purpose of the filter development is to reconstruct terrain surfaces in a complex urban environment, and thus, to support a building extraction process. In an urban area, however, there are some difficulties which make the filters fail to properly classify LIDAR points. These filtering difficulties posed by urban environment are related to the object variety and complexity. In particular, the following objects can be considered as obstacles for the filter to overcome:

- **Very low and small objects** – There are many small and low objects such as vehicles, walls, and low vegetation in urban area. The closer an object is to the terrain surface, the more difficult it becomes for the filter to differentiate the low objects from the terrain.
- **Disconnected terrain** – In an urban area, one often to come across disconnected terrains, which are surrounded by buildings and thus, disconnected from nearby terrains. A courtyard is a good example of disconnected terrain.
- **Discontinuous terrain** – Due to the effects of city development, urban terrain surface commonly shows a high discontinuity. Terrain segments are artificially reconstructed to have sharp ridges.

- **Attached objects** – Some sides of buildings could be attached to nearby terrain surface at the same height. Parts of bridges or gangways are connected to the terrain surface or buildings, while elevated over rivers or roads.

This chapter aims to validate how the filtering rationale given by the RTF filter operates in the urban environment mentioned above. To this end, two sets of LIDAR DSM are chosen, each of which has different terrain characteristics. In the next section, the RTF filter is applied to a sub-urban area using the Shrewsbury data. The result of the reconstructed terrain surface will be investigated with the support of airborne images, digital maps and GPS measurements. Based on the result of the Shrewsbury data, some improvements of the RTF filter will be described in §4.3. Finally, the OEEPE LIDAR DSM is used for the validation of the RTF filter in urban environment. The filtering result will be demonstrated in comparison with manually filtered reference data. An initial development of the RTF has been evaluated by the Working Group III/3 of ISPRS Commission III in the framework of "3D Reconstruction from Airborne Laser Scanner and InSAR Data" (Sithole & Vosselman, 2003b). Some initial results of the RTF filters can be found in Fischer (2002) and Sithole & Vosselman (2003c).

4.2 CASE I: Shrewsbury data

4.2.1 The study area

The study area is part of Shrewsbury in Shropshire in Western England and lies between 351,149 to 352,884 E and 312,047 to 313,276 N in the GB National Grid. A location map and aerial image of the study area are shown in figure 4-1. The dimensions of the study area are 1,735 m in the East-West direction, and 1,229 m in the North-South direction, an area of 2.1 km^2. The land cover of this area consists of forest, residential area, river, and agricultural fields. In the study area, elevation values range from about 43m to 75m. The lowest terrain point can be found in the flood plain of the River Severn, and highest terrain points are located in the South-West corner of the study area. The River Severn, which runs through the middle of the study site, subdivides it into two parts with different land covers and terrain characteristics. The Western part, sloping towards the river, shows a mixture of dense urban area, railway embankment, meadows, single trees, groups of trees, shrubs and a few agricultural fields with hedges. The Eastern part is characterized by fairly flat terrain with a

significant low amount of non-terrain features, which contains a railway embankment with trees and shrubs, a few hedges and a major road with roundabout. A dense forest is located along the river embankment.

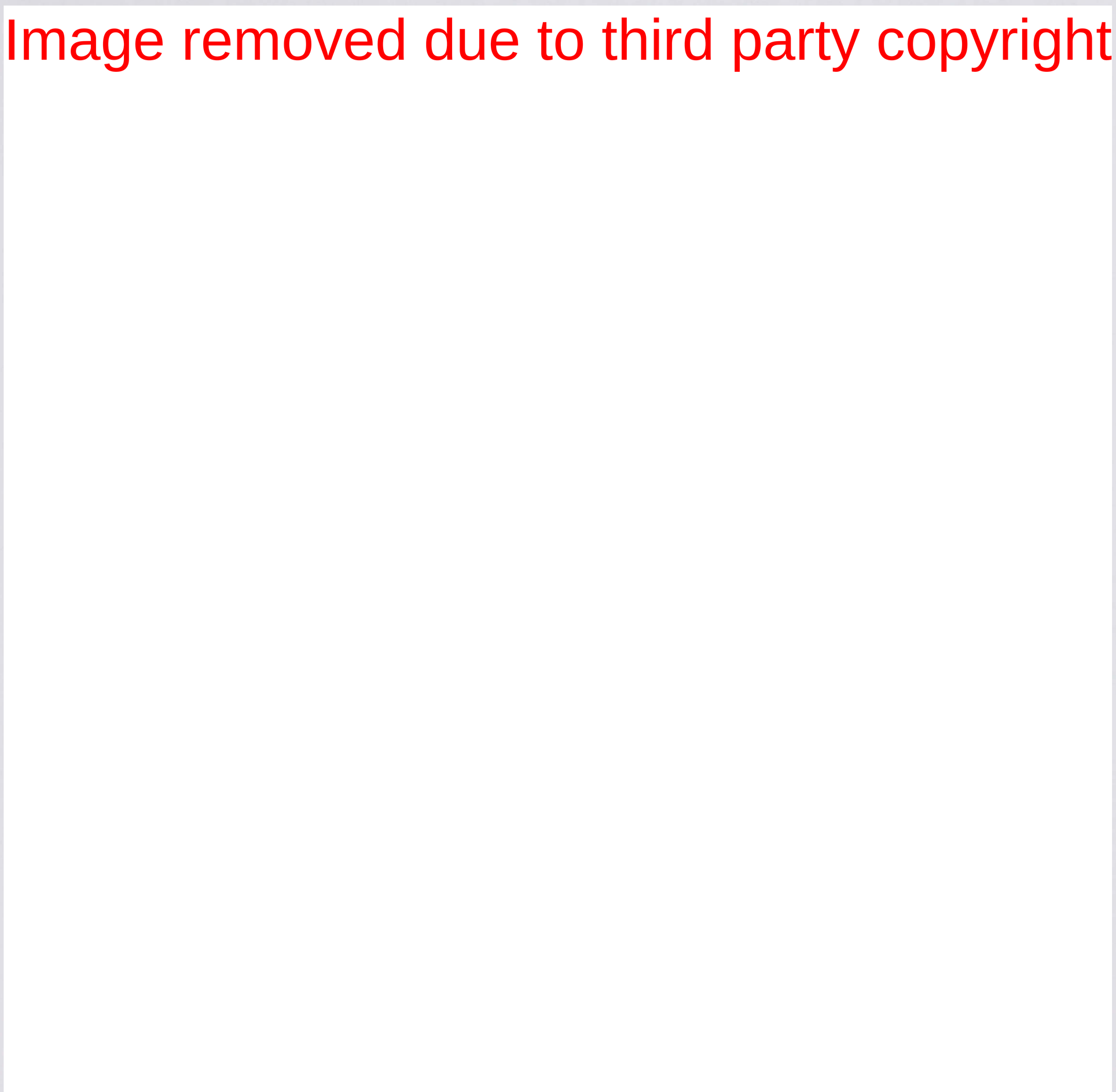


Figure 4-1. Study area of Shrewsbury (Fischer, 2002)

4.2.2 Data set used

4.2.2.1 LIDAR DSM

Figure 4-2 shows a LIDAR DSM covering the Shrewsbury study area, which was provided by the Environment Agency of the United Kingdom. The LIDAR DSM was acquired by the last returns of the Optech ALTM 1020 sensor in March 1999. The characteristics of the Shrewsbury LIDAR DSM are described in table 4-1. The parameters of the Optech ALTM 1020 are summarized in table 4-2. The vertical accuracy of the sensor is stated by the manufacturer with ± 0.11m to ± 0.25m in the root mean square error (RMSE). The data has been converted from UTM/WGS84 to OSGB36 (plan) and OSD Newlyn (height) using OSTN'97 for X,Y and OSGM'91 for Z which potentially adds an error of 10 cm to the data (Fischer, 2002). Figure 4-3 shows the Shrewsbury LIDAR DSM in 3D with different perspective views.

Figure 4-2. Shrewsbury LIDAR DSM

Table 4-1. Data characteristics of the Shrewsbury LIDAR DSM

Parameters	Values
Total number of LIDAR points	561,549 (points)
Point density	0.26 (points/m^2)
X range	351149.42 ~ 352884.18 (metre)
Y range	312046.97 ~ 313276.37 (metre)
Z range	42.93 ~ 95.76 (meter)

Table 4-2. Sensor parameters of the Optech ALTM 1020 (Fischer, 2002)

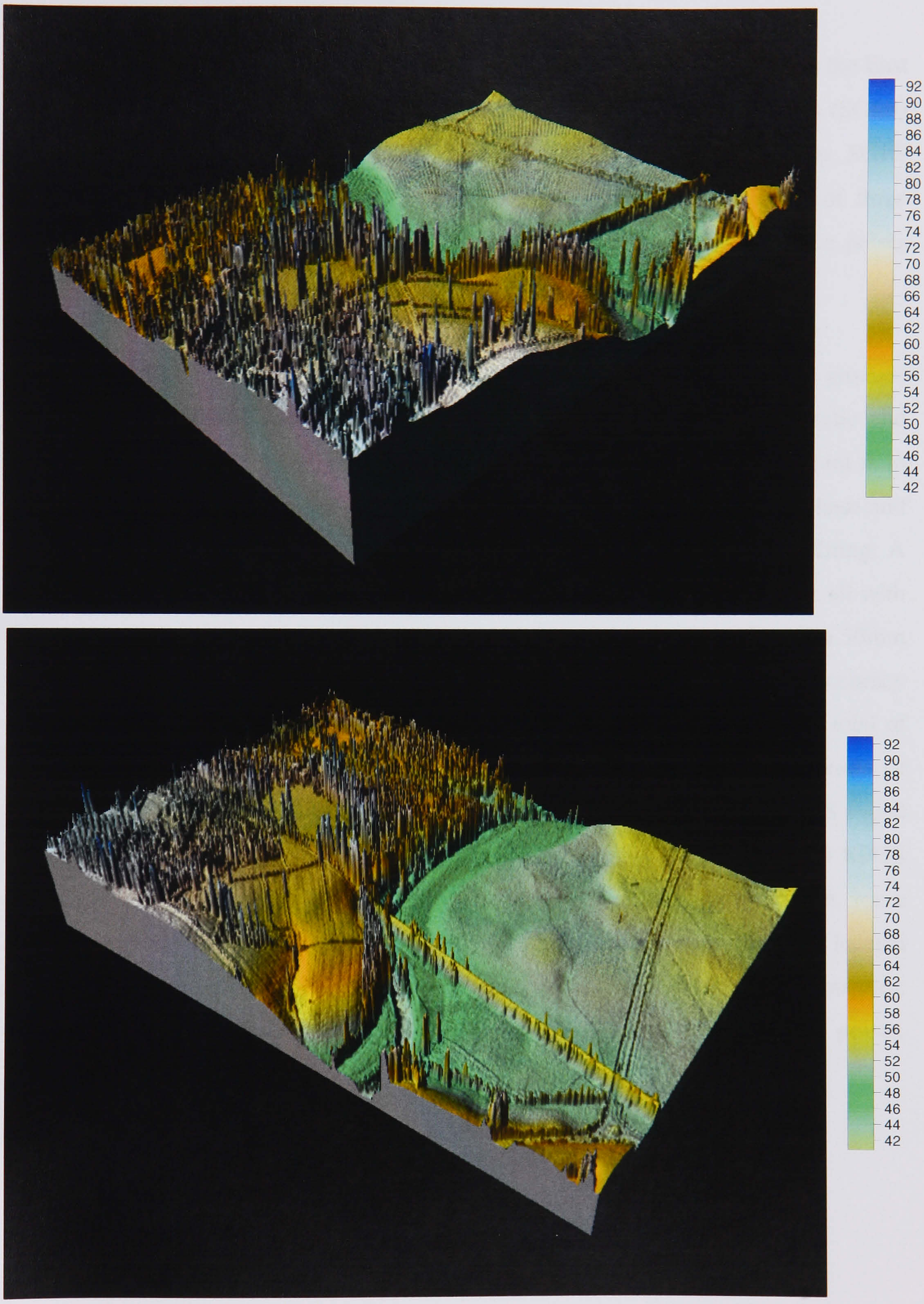


Figure 4-3. Shrewsbury LIDAR DSM with different perspective views

4.2.2.2 GPS data

Two different types of GPS (Global Positioning System) data were observed on the East side of Shrewsbury by UCL as part of the NextMap project; static GPS points (SGPS) and kinematic GPS profiles (KGPS) (Ziebart & Adhya, 2002). For current study, SGPS data was used as GCPs (Ground Control Points) for the orthorectification of three overlapping aerial photographs, while KGPS data was used as terrain reference points for quantifying an accuracy assessment of DTM.

A central reference station was established with a precision of ± 25mm in the grounds of North Farm, Eaton Mascot, which location is suitable to support rapid static and kinematic GPS positioning (Ziebart & Adhya, 2002). Once established this point is to be continuously occupied by a GPS receiver while a roving receiver collects phase and pseudorange data at the points selected for photogrammetric control and profiling. A total of 16,000 KGPS points were recorded using epoch rates of 10Hz and 1Hz all with successful ambiguity resolution. The vertical precision of these points was at the 50mm (2σ) level with respect to the reference station coordinates, and the horizontal accuracy was given with ± 0.1m (Fischer, 2002). With the same accuracy noted above, a total of 12 SGPS points were measured as photogrammetric control points on the basis of providing a good geometric spread of points while avoiding sites with poor GPS signal coverage and high multipath. Among the observed GPS points, a total of 4,489 KPGS points are common to the study area, which are grouped into three profiles for a separate accuracy assessment of DTM quality. Of 12 control points, 8 SGPS are located within the study area, and 5 SGPS points are used as GCP points for photogrammetric bundle adjustment. Figure 4-4 shows the distribution of SGPS and KGPS points, located in the study area.

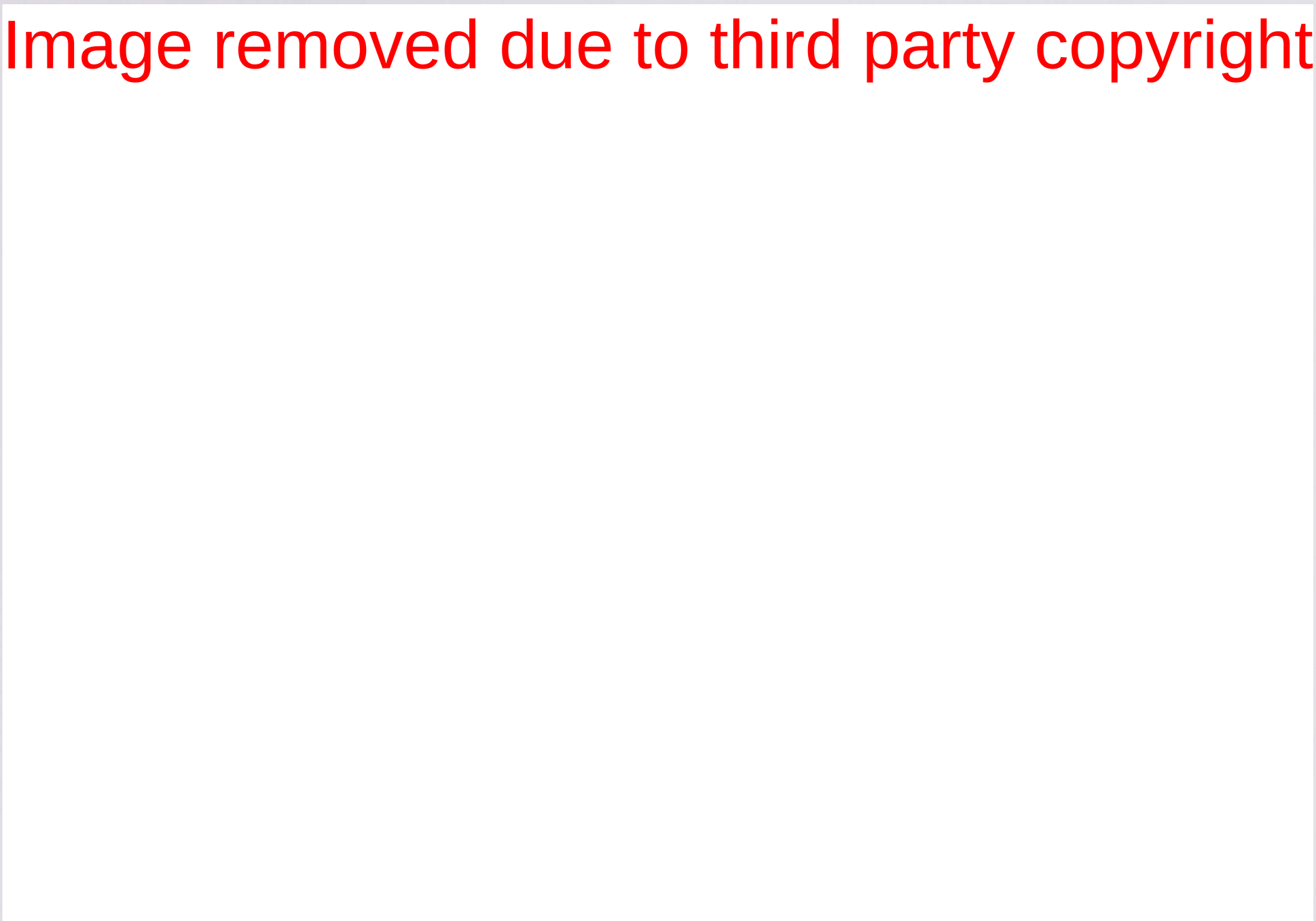


Figure 4-4. GPS measurement in Shrewsbury study area; red triangles represent SGPS measurements, while yellow dot for KPGS profiles (Fischer, 2002)

4.2.2.3 Orthorectified aerial photograph

Three overlapping aerial photographs were acquired on the 1st of November in 2001 from a flying height of about 1,200m using RC30 frame camera, with a focal length of 153.6mm (Fischer, 2002). Each image overlaps its adjacent one by about 60%. These aerial photographs were scanned at a resolution of 21 μm, which corresponds to the map scale of 1: 7500. From these aerial images, orthorectified imagery was generated to support a visual inspection of DTM quality through established photogrammetric methods. For generating the required orthophoto, a photogrammetric stereo-model from aerial photography was provided by UCL. The model had been generated using Socet Set DPW (Digital Photogrammetric Workstation) (Papadaki, 2002). An interior orientation was calculated with given camera calibration certificate and known sensor model, followed by exterior orientation with 5 SPGS points and 17 tie points measured by Socet Set. Final triangulation accuracy was measured by Papadaki (2002), with the horizontal RMSE was 0.080 and 0.067 metre in (x,y), and the vertical RMSE is 0.023 metre. Once the stereo-model was set up, a DEM (Digital Elevation Model) with 1m

resolution was produced, adapting a rolling dense strategy, which is one of the pre-designed area-based matching techniques provided by Socet Set. Finally, orthorectified imagery with no relief distortion was generated by the DEM. Figure 4-5 shows that the generated orthophoto is successfully overlaid with the OS Landline.Plus®.

4.2.2.4 Digital map

The Ordnance Survey (OS) Land-Line.Plus® data was used for a visual inspection of DTM quality. The OS Land-Line.Plus® data provides detailed topographic information of man-made and natural features at three basic scales; 1:1250 (urban areas), 1: 2500 (rural areas) and 1: 10000 (moorland areas) with full coverage of Great Britain. For the current study, data at the largest scale and upgraded in May 2001 were used. The Land-Line.Plus® data, supplied in NTF format, were converted into ArcView shapefiles by means of ESRI's UK Map Manger 6.2. The absolute horizontal accuracy of the 1: 1250 vector data is less than ± 0.4 m. Figure 4-5 shows the OS Land-Line.Plus® digital map overlaid with the orthorectified aerial image over the Shrewsbury study area.

Figure 4-5. Ordnance Survey Land-Line.Plus® overlaid with the orthorectified aerial image in Shrewsbury study area (Fischer, 2002)

4.2.3 Automatic DTM generation

The RTF filtering algorithm was applied to the LIDAR DSM of figure 4-2 for the generation of the DTM. Before the filter actually performs, data pre-processing was performed. First, a thinning process was applied to the DSM. This process aims to remove too closely spaced LIDAR points from neighbouring points to properly construct a TIN. Considering the precision of Delaunay Triangulation algorithm implemented, a point was removed if the planimetric distance between the point and its neighbouring point is less than 1 cm. This thinning process reduced 561,549 points of the original LIDAR points down to 503,212 points.

With the thinning procedure, a tiling method was also applied to the LIDAR DSM in order to speed up the filtering process. A creation of LIDAR DSM tile is a very simple procedure, where the entire DSM can be split into a number of tiles which are all the same size. This resulted in the creation of 20 DSM tiles as 0.35 km x 0.3 km was selected for the size of a tile, the last column of which were slightly smaller than the rest of tiles. The tile size selected must be large enough not to be completely nested inside a building rooftop. This tiling approach is useful for the filter to be independently applied to each tiled DSM in parallel so that the overall computational cost can be reduced. However, a tiling error might degrade the constancy of the filtering result across the boundaries between adjacent tiles. To cope with this problem, the main process of the RTF filter, downward terrain fragmentation (DTF) and upward terrain fragmentation (UTF), were applied to each tiled DSM, but the post-processing was applied to an integration of results extracted from all the tiles so that the classification result of LIDAR points can be reinvestigated across the tiling boundaries.

Once the data preparation was finished through the aforementioned thinning and tiling process, the RTF filter was applied to each tiled DSM data. As described in §3.4.1, an initial terrain model was generated for each tile. Then, the DTF process started over all the initial terrain models and found 1,639 on-terrain points. For the UTF process, a number of parameters were pre-determined and table 4-3 shows these parameters.

Table 4-3. RTF filtering parameters for the Shrewsbury LIDAR DSM

Parameter	Value	Process	Determination
δ_h	1 (meter)	Upward terrain fragmentation	Pre-defined
α	0.1	Upward terrain fragmentation	Pre-defined
β	45 (degree)	Upward terrain fragmentation	Pre-defined
δ_{PC}	0.3	Region-based post-processing	Pre-defined

As a heuristic height threshold to discriminate between on- and off-terrain points over a flat or plane terrain surface, δ_h was selected as 1 metre, where a LIDAR point with 1 metre from a plane terrain surface is classified as on-terrain point, otherwise off-terrain point. Considering the point spacing of 3 meter of the Shrewsbury dataset, the value of δ_h imply that the filter can detect a terrain slope variation of about 18° or even more if a local point spacing that the filter is applied is smaller than 3 metre. Other heuristic parameters were chosen for α and β used in the MDL model selection defined in Eq. 3-22. These vales are heuristic parameters for separating on-terrain points from off-terrain points in terms of slope angle. For the current experiment, both of α and β were selected as 0.1 and 45° respectively.

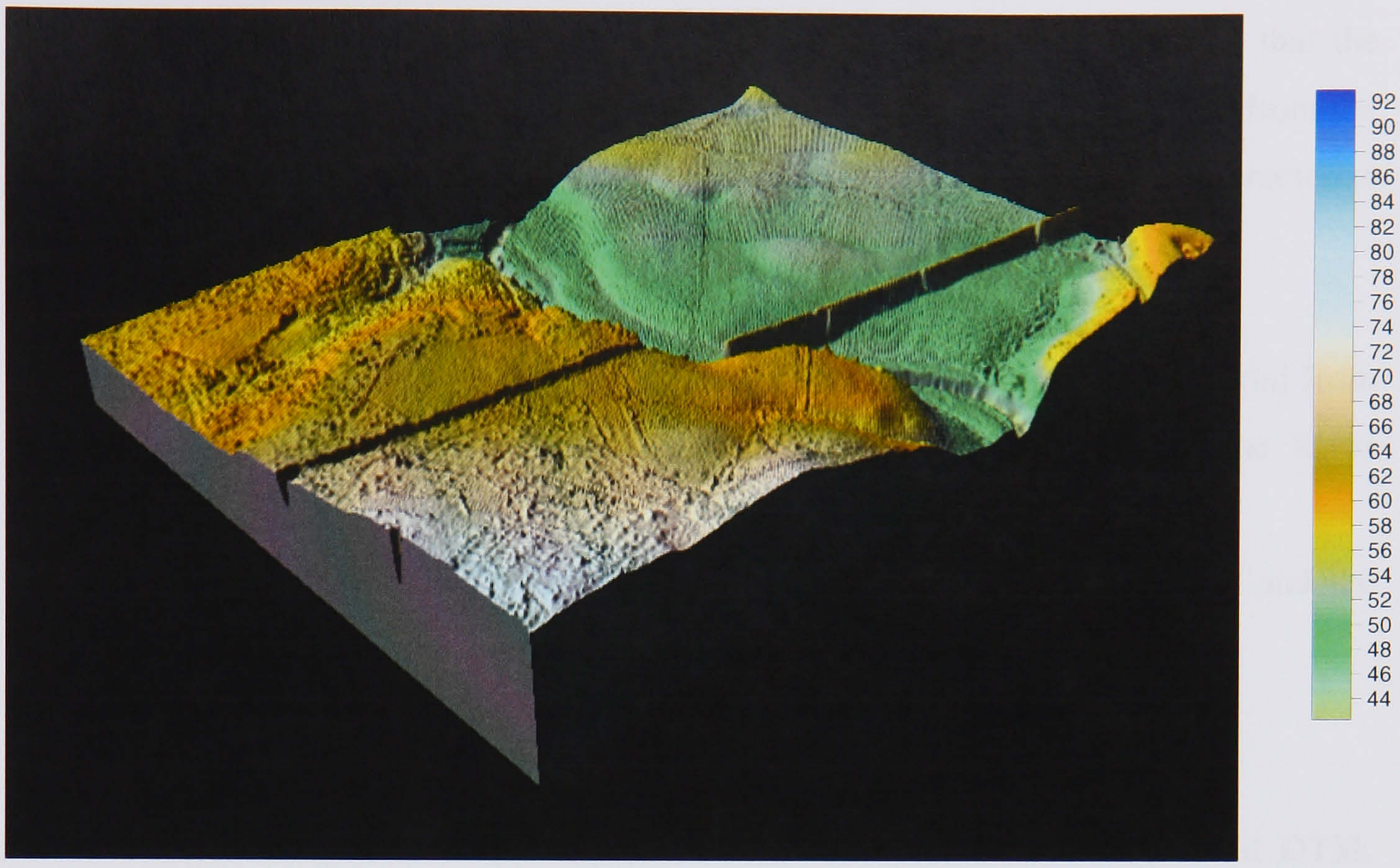
With these parameters mentioned above, the UTF filter was applied to all the DSM tiles and recognized 445,338 LIDAR points as on-terrain points and 57,874 points as off-terrain points. After the UTF process was finished, the classification results extracted from all the tiles were integrated as one dataset, over which the post-processing was applied. As for the region-based post-processing described in §3.4.4.2, the δ_{PC} of 0.3 was chosen in Eq. 3-26 to remove the Type II errors of an attached building. With this value, if more than 30% of boundary points of a segmented on-terrain region show obvious height discontinuity, i.e., higher than δ_h from neighbouring points, the region is corrected as an off-terrain one. Since the Shrewsbury test dataset does not contain a complex situation where a building is attached to a terrain surface, the region-based post-processing did not change the previous result of UTF. The final step of the RTF filter is the point-based post-processing described in §3.4.4.3. This process corrects mis-classified off-terrain points as on-terrain ones. This process re-classified 150 off-terrain

points as on-terrain points and resulted in the detection of 445,488 on-terrain points and 57,724 off-terrain points. Table 4-4 shows the final result of the RTF filter applied to the Shrewsbury dataset. Figure 4-6 shows a hill-shaded result of filtered DTM. Figure 4-7 also shows the DTM result of the Shrewsbury dataset reconstructed by the RTF filter with different perspective views.

Table 4-4. RTF filtering result of the Shrewsbury LIDAR DSM

Processing step	Results (points)	
	on-terrain	off-terrain
Downward terrain fragmentation (DTF)	1639	501573
Upward terrain fragmentation (UTF)	445338	57874
Region-based post-processing	445338	57874
Point-based post-processing	445488	57724
Total	503212	

Figure 4-6. RTF filtering result over the Shrewsbury LIDAR DSM



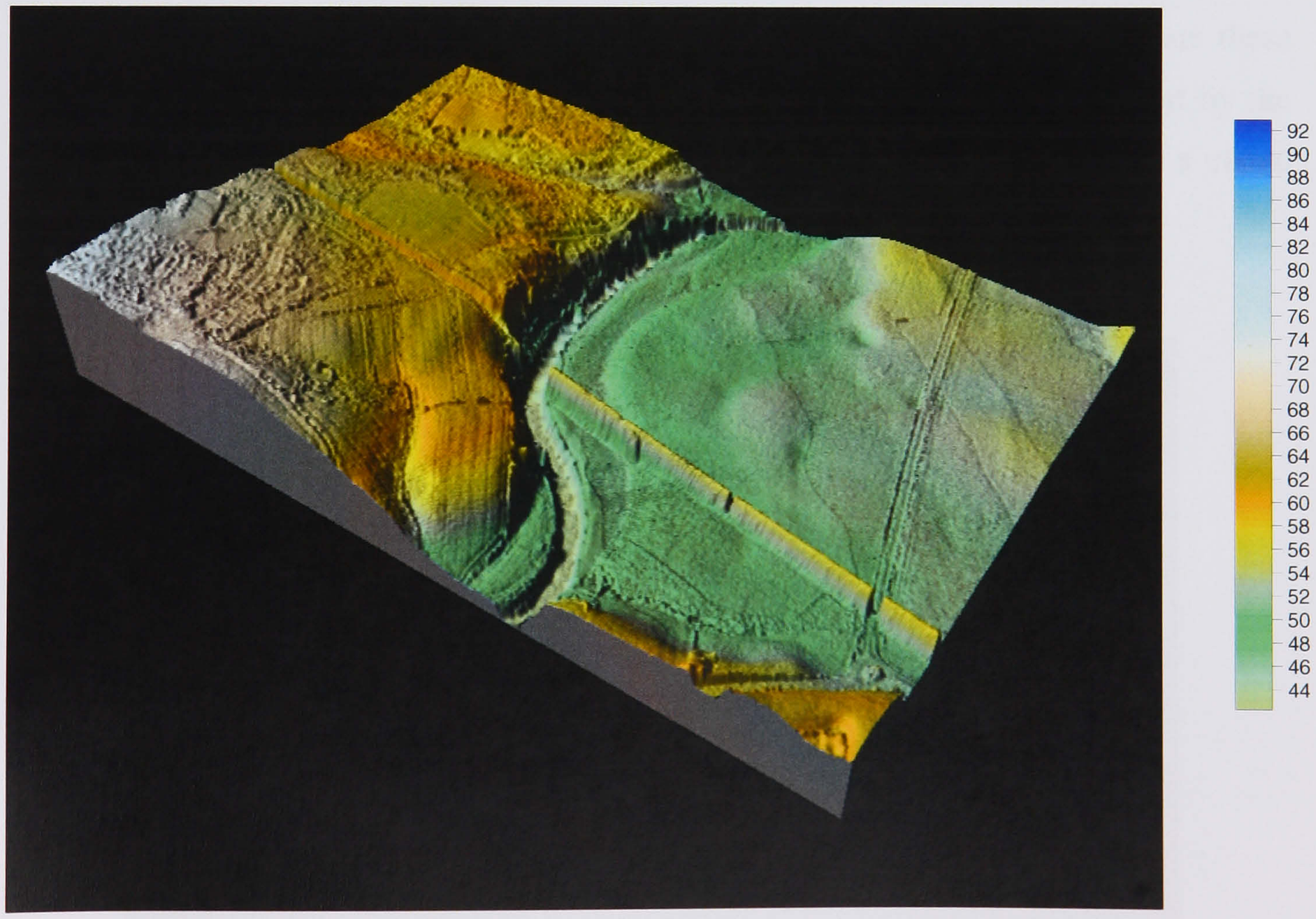


Figure 4-7. RTF filtering result over the Shrewsbury LIDAR DSM with different perspective views

4.2.4 Visual quality assessment

From a visual examination of figure 4-6 and figure 4-7, it can be concluded that the overall performance of RTF filter successfully filtered off non-terrain features from the Shrewsbury dataset. In particular, the following aspects can be addressed to characterize the RTF filter:

- Most off-terrain features such as trees and buildings were clearly removed from residential areas and dense forest areas near the embankment of the River Severn regardless of terrain slope variations.
- Significant geomorphologic features such as embankments of railway and the River Severn, where sharp slope transitions occur, are preserved.
- Railway bridges across the river and motorways are removed.

However, some drawbacks of the filter can be also observed. In the result of DTM, some peculiar spots comprising one or two off-terrain points can be seen. Also, part of hedges and low bushes still remained in the reconstructed DTM. To investigate these drawbacks, four sub-areas were selected from the Shrewsbury dataset depicted by the red boxes of figure 4-8. With these sub-areas, more detail analysis relying on a visual investigation will be discussed in following section.

Figure 4-8. Sub-areas selected for assessing the quality of the RTF filtering result

Area-1

(a) ariborne imagery

(b) LIDAR DSM

(c) filtered DTM

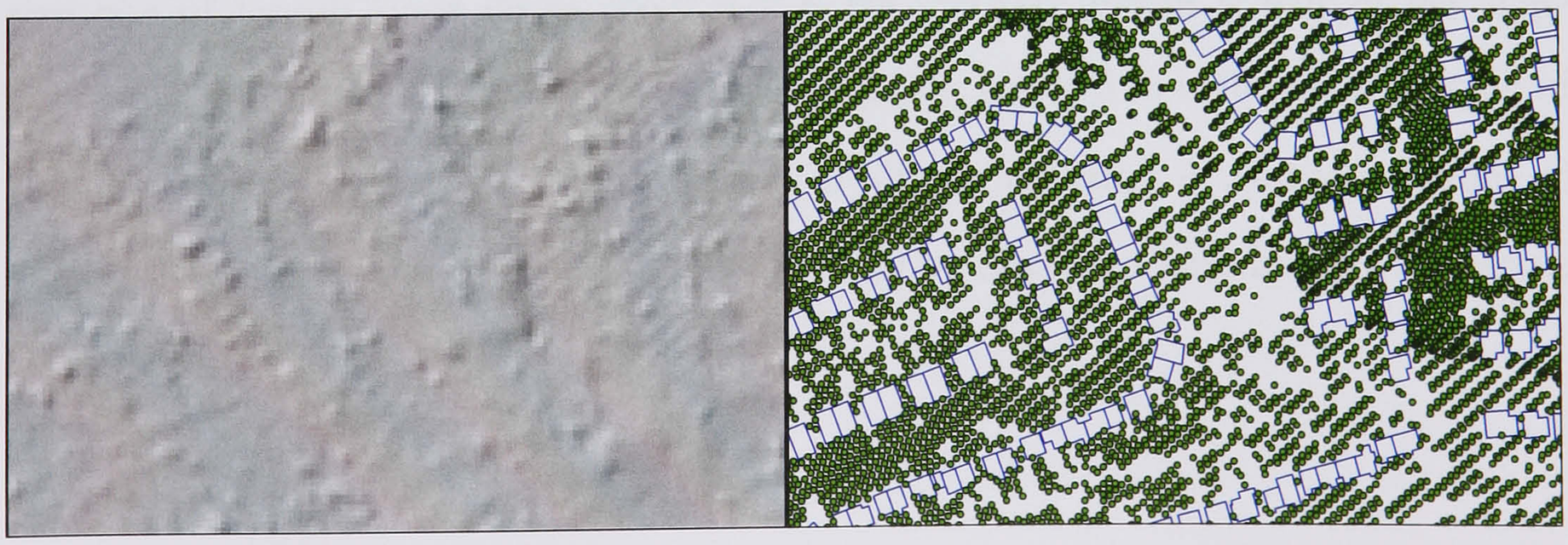
Figure 4-9. RTF filtering results of Area-1 in Shrewsbury study area

As can be shown in figure 4-9 (a), Area-1 is covered by different landforms. Some residential houses can be seen in the bottom of the area, where low objects such as shrubs and cars are located near to houses. The rest of region is a fairly flat meadow, where comparatively large trees are located. In the middle of the area, some knolls whose slopes smoothly change, and a group of trees and bushes can be found. Figure 4-9 (b) and (c) presents an original LIDAR DSM and filtered DTM respectively. Obviously, most of residential houses and high trees were neatly removed after the RTF filter was applied to figure 4-9 (b). The round shape of the knoll appearing in the middle of the area was well preserved in the reconstructed DTM. However, a few of granule-

like artifacts are scattered in the residential area. One of these artifacts can be seen as a red circle in figure 4-9 (c). It is not clear where these noises come from due to the poor resolution of LIDAR data used and a time gap between acquisitions of airborne imagery and LIDAR data. After a manual inspection, it was found that these noisy artifacts are made up of only one or two LIDAR points and the height difference between these points and their neighbouring points is of 50~80 cm. It suggests that those artifacts remaining in the DTM came from low off-terrain features such as small trees, shrubs and cars that the filter did not properly filter them off. For observing these artifacts in more detail, other residential area with larger extent was selected as Area-2.

Area-2

(a) airborne imagery (b) LIDAR DSM



(c) filtered DTM (d) on-terrain points overlaid with building layer

Figure 4-10. RTF filtering result of Area-2 in Shrewsbury study area

Figure 4-10 (a) shows a sub-region of Area-2, which is mainly covered by a number of residential houses over fairly flat terrain. In Area-2, small trees are also located nearby houses and low objects can be seen on the streets. Some large trees are found in the top and the middle of the scene. From a LIDAR DSM of figure 4-10 (b), a DTM was reconstructed by RTF filter as presented in figure 4-10 (c). On-terrain points classified by the filter were overlaid with the Land-Line.Plus® ground plan vectors provided by the Ordnance Survey. It can be clearly seen in figure 4-10 (d) that most of on-terrain points were correctly classified as they were located outside of building boundaries. However, similar to the case of Area-1, a number of granule-like artifacts were not well removed and a rugged surface appears in the reconstructed DTM. It was also confirmed that only one or two on-terrain points comprise each artifact and its height difference with the neighbouring points is usually larger than 50 cm, but less than 1 m. As in the case of Area-1, it implies that the noisy artifacts were caused by low objects such as small trees and cars located in nearby houses or on the street, which were not properly removed by RTF filter. This can be understood as δ_h of 1 metre selected for differentiating off-terrain points from on-terrain ones is too high to filter off the low objects. Thus, small shrubs, hedges and other surface features, of which vertical extent is below 1m, remain in the DTM. This problem can be solved by the use of lower δ_h. However, too much lowered δ_h may make the filter "under-adjusted" to terrain slope variances. For instance, if δ_h is reduced to 50 cm, the RTF can be adjusted to the variance of terrain slope up to about 9.5° = $\tan^{-1}$ (0.5/3), which is half of the slope variance achieved by δ_h of 1 m (18.4° = $\tan^{-1}$ (1/3)), where 3 m is the point spacing of the Shrewsbury dataset. Therefore, instead of simply lowering δ_h, other solutions to remove low objects must be considered. This will be discussed later in §4.3 with some suggestions to improve the RTF filter.

Area-3

As can be seen in figure 4-11 (a), Area-3 is covered by agricultural fields and meadows. The western part of Area-3 is much higher than the Eastern part and the height difference between two areas is about 15 m. The River Severn flows from the top to the lower right of the scene. A dense forest can be seen along the river embankment. A railway embankment runs through the middle of the scene, where shrubs and trees are located on the side of the railway embankment. A railway bridge can be also seen at the

centre of the scene. Although comparatively large discontinuities exist in between the Eastern and the Western side of the river, figure 4-11 (c) shows that the RTF filter successfully generated the DTM from LIDAR DSM of figure 4-11 (b). In particular, a geomorphologic characteristic of the embankments of the river and the railway were well preserved. Moreover, single trees, or a group of trees on the side of the railway were clearly filtered off, while a terrain surface underneath a dense forest was nicely reconstructed.

(a) orthorectified airborne imagery

(b) LIDAR DSM (c) filtered DTM

Figure 4-11. RTF filtering result of Area-3 in Shrewsbury study area

However, as in the cases of Area-1 and Area-2, a few artefacts caused by low objects can be observed near the river embankment (see red circle of figure 4-11 (c)). It was confirmed that one of those noisy points was used as a virtual corner point for

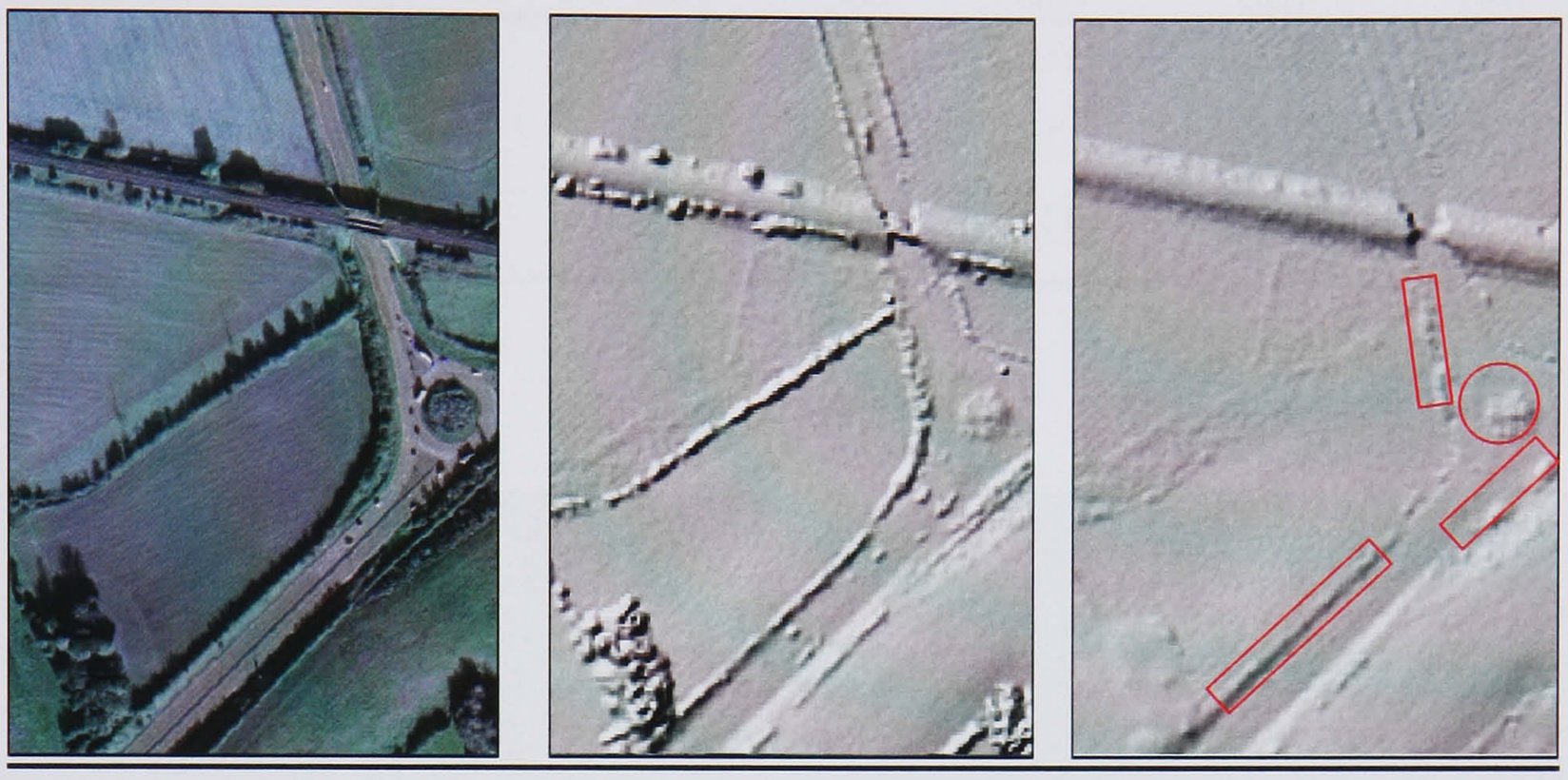
generating an initial terrain model of underlying tiled sub-region and the corner point is unfortunately located on top of the off-terrain feature. In this case, the corner point produces some noise in the generation of DTM since an on-terrain label is explicitly assigned to the corner point with no validation. Although region-based post-processing was applied in order to correct the misclassification of the corner point, it failed because the low object, over which the corner point is located, is not well isolated from its surroundings. However, the region-based post-processing can remove the error if the point is located on the top of buildings which are distinguishable from surrounding terrain with an obvious height discontinuity.

In addition, hedges within the agricultural area partly remain in the DTM (see the red box in figure 4-11 (c)). The overall height of the hedge falls in a range of 1.5 m ~ 1.8 m. The Shrewsbury DSM was acquired by the last pulse of the Optech's LIDAR system. Some laser pulses may penetrate towards the ground underneath the hedge, rather than be reflected from the top of the hedges. Thus, the last pulse acquisition makes the height discontinuity of the hedge less distinguishable. For this reason, parts of the hedges can be recognized as a low object, which are labelled as on-terrain points by the selected δ_h of 1 m. However, the artifacts caused in the area formerly covered by the hedge are localized within a small region. This is a promising property because a simple smoothing technique can remove those noises.

Area-4

Figure 4-12 shows a DTM reconstruction result of Area-4. Similar to the case of Area-3, a group of large trees in the agricultural area was clearly removed. The railway embankment was well preserved, while single trees and shrubs on the side of the railway embankment were removed and the railway bridge crossing the motorway was filtered off. However, the hedges located along the motorway depicted as red boxes in figure 4-12 (c) were not perfectly removed due to the penetration of the last laser pulses. Also, the roundabout on the motorway remains as part of the terrain model (see the red circle in figure 4-12 (c)). The maximum height difference between the centre of the roundabout and surrounding terrain was roughly measured as about 3 m, but the height transition between them occurs within less than 1 m. Thus, the roundabout cannot be isolated as an off-terrain feature, but continuously classified as on-terrain points as the

RTF proceeds, since some LIDAR points located on the boundary of the roundabout falls within the selected δ_h of 1 m.



(a) airborne imagery (b) LIDAR DSM (c) filtered DTM

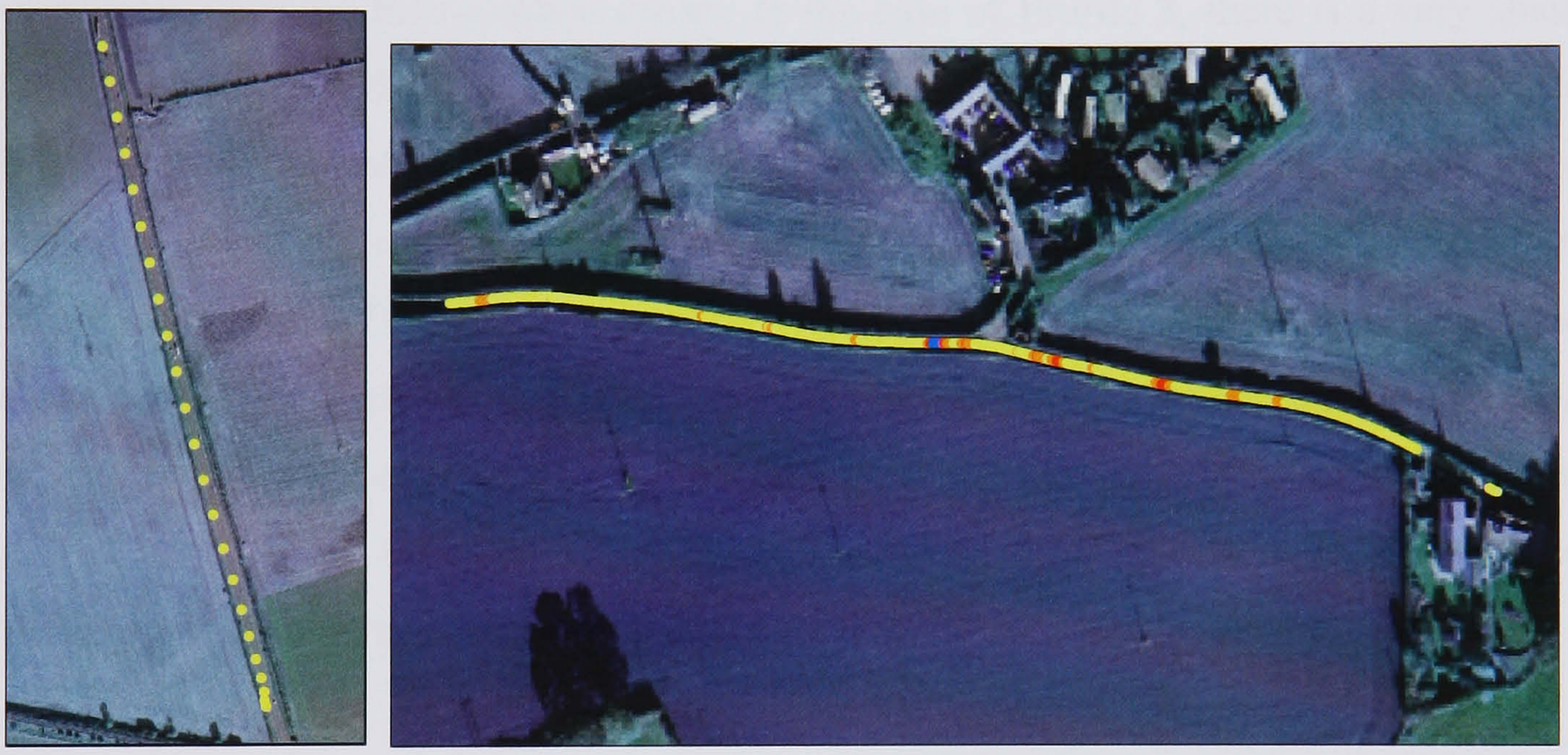
Figure 4-12. RTF filtering result of Area-4 in Shrewsbury study area

4.2.5 Quality assessment by KGPS profiles

Three KGPS profiles described in §4.2.2.2 were used for a vertical accuracy assessment of the DTM generated by the RTF filter. To this end, on-terrain points labelled by the filter were interpolated by means of the inverse distance weight (IDW) algorithm, which results in a gridded terrain surface. Then, KGPS profiles were projected onto the interpolated terrain surface, from which vertical differences were measured. Table 4-5 describes the result of the vertical accuracy of the filtered Shrewsbury DTM in comparison to KPGS profiles. The RMS residuals between the KGPS profiles and the filtered DTM are overlaid with orthorectified airborne imagery in figure 4-13.

Table 4-5. Vertical accuracy assessment of the filtered Shrewsbury DTM in comparison to KPGS profiles

KGPS	Points	Min (m)	Max (m)	Mean (m)	Std Dev. (m)	RMSE (m)
Profile 1	25	-0.115	0.265	0.095	0.102	0.139
Profile 2	1050	-1.091	0.180	-0.110	0.178	0.209
Profile 3	1334	-0.747	0.189	-0.065	0.080	0.103



(a) Profile 1

(b) Profile 2

(c) Profile 3

Figure 4-13. RMS residuals in height between the filtered Shrewsbury DTM and KPGS profiles

As can be seen in table 4-5, the RMS residuals between the KGPS profiles and the filtered DTM are about 0.15 m in average. The Profile 2 shows the biggest error. This may be caused by the hedges along the road, which could interfere with GPS signals, leading to the degradation of the ranging accuracy. An influence of the interpolation error was investigated. A total of 71,457 on-terrain points around KGPS profiles were collected (see table 4-6). Similar to the error measurement using KGPS profiles, vertical differences were measured between on-terrain points and the interpolated terrain surface. As can be shown in table 4-6, the interpolation error of the IDW algorithm is measured as about 0.12 m RMSE in height. The RMSE errors of Profile 1 and Profile 3 are almost compatible with the interpolation errors. In the case of Profile 3, there is a very small difference, in which the KGPS error is higher than the interpolation error in about 0.1 m RMSE. As mentioned above, ranging blunders caused by the hedges may lead to this small amount of error discrepancy. Thus, the accuracy assessment using KGPS profiles suggests that the RTF filter reconstructs terrain surface with very high quality, and produces random errors, whose magnitude is compatible to the one of the interpolation error. However, the error analysis using KGPS survey is limited to very smooth surfaces such as roads. In §4.4, further accuracy assessment of the RTF filter will be described over regions with wider range of terrain characteristics.

Table 4-6. Interpolation error assessment of the Shrewsbury DTM with respect to KPGS profiles

KGPS	Points	Min (m)	Max (m)	Mean (m)	Std Dev. (m)	RMSE (m)
Profile 1	65535	-0.928	1.4586	-0.002	0.130	0.130
Profile 2	4651	-0.838	0.968	0.000	0.119	0.119
Profile 3	1271	-0.473	0.716	-0.001	0.096	0.096

4.3 Improvement of RTF filter

As described in §4.2, the filtering result of the Shrewsbury data shows that the RTF filter has a drawback because it does not properly remove low objects such as small cars and bushes. This is caused when a height threshold of δ_h is selected so high that low objects are located within a height buffer created by δ_h from a plane terrain model (PTM). In this case, the low object points serve to classify the PTM as the "continuous terrain". As a result, the upward terrain fragmentation (UTF) is triggered, and the low object points are finally labelled as on-terrain points. This problem can be solved by simply selecting a lower value of δ_h. However, as the δ_h is lowered, the filter becomes insensitive to detect the terrain slope variation, and it may fail to find the coexistence of different terrain slopes within the lowered height buffer. Figure 4-14 illustrates a situation, in which a small tree is properly classified as the off-terrain object by δ_h, but the points located on the sloped terrain surface are not detected by the same δ_h. In consequence, the sloped terrain is misclassified as an off-terrain object.

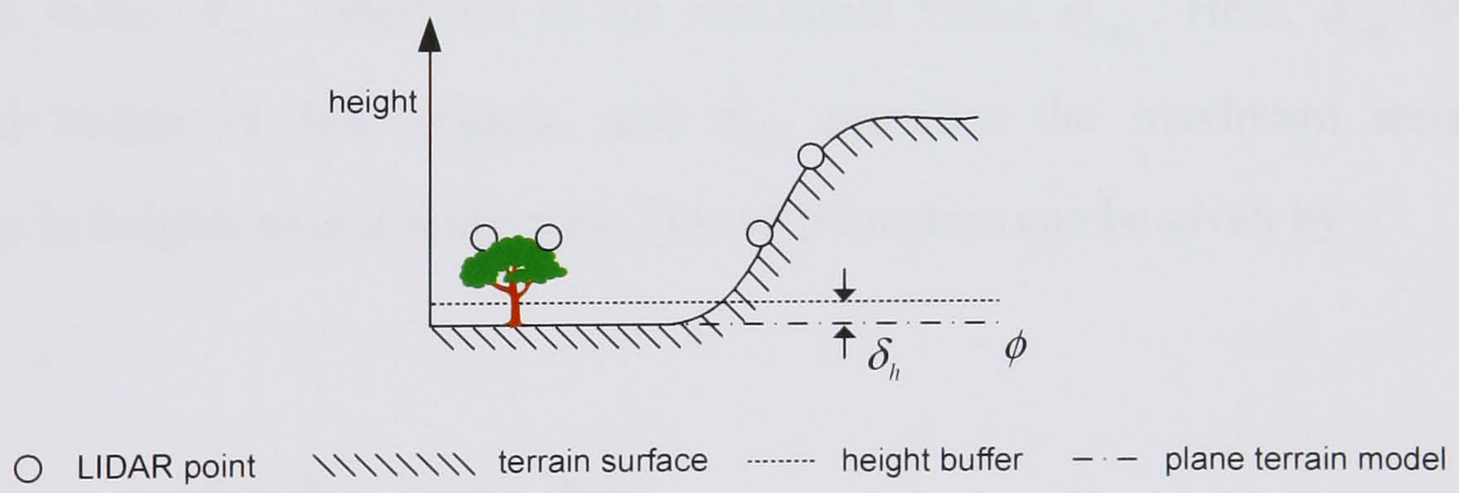


Figure 4-14. Illustration of triggering the UTF process by a low δ_h

The aforementioned problem is caused by the fact that δ_h is eager to simultaneously detect both off-terrain points and terrain slope variations. Thus, this problem can be resolved by separating both roles of the δ_h. To this end, it is assumed that a large variation of the terrain slope may exist within a PTM of ϕ with larger extent, while a degree of the terrain slope variation may decrease within a smaller ϕ. Based on this assumption, δ_h is changed depending on the size of ϕ. That is, over a ϕ with large

extent, the filter focuses on detecting the terrain slope variation by selecting a higher value of the δ_h (see figure 4-15(a)), while the δ_h becomes lowered to filter off the low objects as underlying terrain surface is more finely fragmented (see figure 4-15(b)).

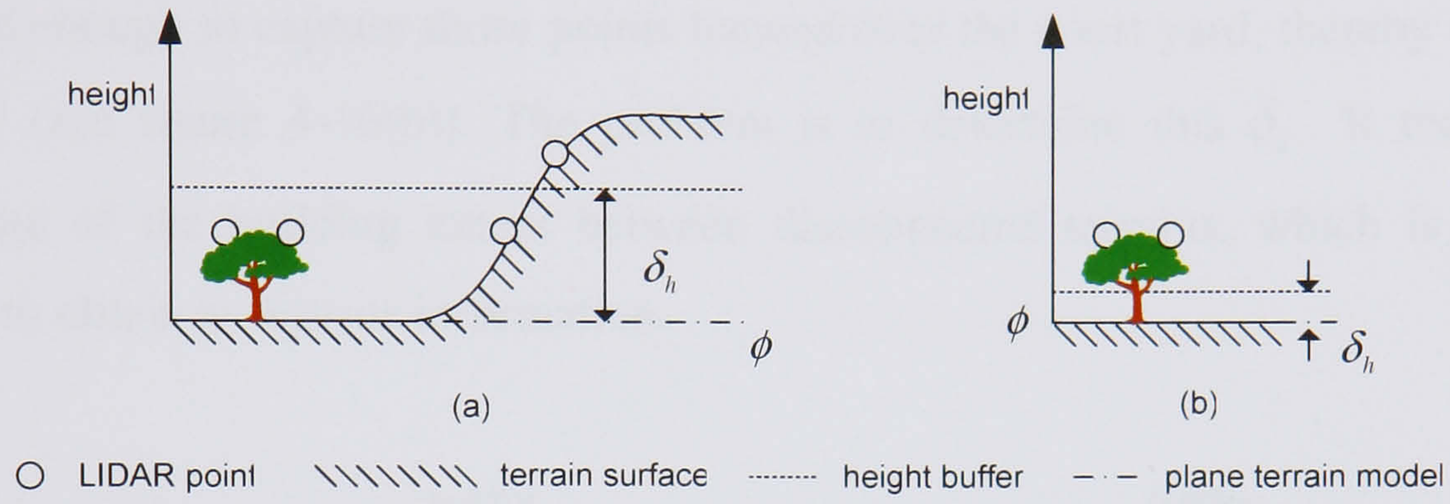


Figure 4-15. Illustration of triggering the UTF process by selecting δ_h depending on the size of underlying local terrain model

A simple step function is adopted for determining the δ_h depending on the size of ϕ. A maximum size of ϕ is defined as $\phi_{\max}$, which is determined as the largest distance of three lateral sides of ϕ. If $\phi_{\max}$ is less than certain threshold, κ, the δ_h is selected as its minimum value, $\delta_{\min}$, otherwise as the maximum value, $\delta_{\max}$. Here, $\delta_{\min}$ defines the minimum height of low objects, and $\delta_{\max}$ provides the maximum terrain slope variations in heights over a study area. This step function can be given by

$$\begin{cases} \delta_h = \delta_{\min} & if\ \phi_{\max} \le \kappa \\ \delta_h = \delta_{\max} & if\ \phi_{\max} > \kappa \end{cases} \tag{4-3}$$

An additional improvement of the RTF filter was considered. In an urban environment, the terrain surface often shows a large slope discontinuity. Over a region where a large terrain discontinuity exists, the UTF could be terminated so early that some terrain segments are classified as off-terrain objects. Figure 4-16 (a) illustrates an example of a disconnected terrain, where a court yard is encompassed by buildings. In this case, with

the current value of ϕ, this is classified as the "discontinuous terrain" since the model is constructed between two terrain segments with a large height difference. Thus LIDAR points located over the courtyard could fail to be captured by δ_h, and the courtyard is recognized as an off-terrain object though its height is similar to the one of nearby terrain surfaces. To reconstruct the courtyard as the terrain surface, δ_h must be increased enough to capture those points located over the court yard, thereby triggering the UTF (see figure 4-16(b)). The problem is to determine this δ_h. It requires the knowledge of the building extent between disconnected terrains, which is normally difficult to obtain as *a prior* information.

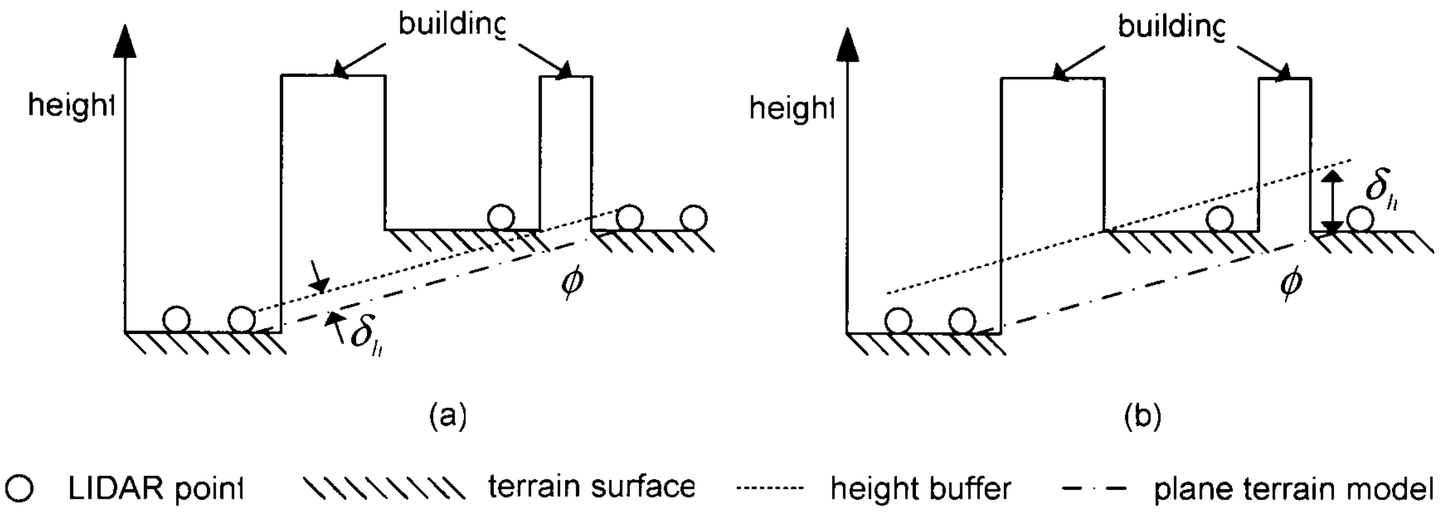


Figure 4-16. Illustration of triggering the UTF process over a disconnected terrain; (a) the UTF process is terminated by a low δ_h; (b) the UTF process is triggered by an increased δ_h

To overcome this problem, a method to classify the "disconnected terrain" is required. This can be done by examining those points whose heights are similar to the one of nearby terrain surfaces. If a local terrain model of ϕ is classified as the "discontinuous terrain", a height buffer is differently generated for the UTF over ϕ. The height buffer is generated as a horizontal plane, which height is higher than a maximum height of the ϕ by the amount of δ_h (see figure 4-17). The ϕ will be classified as the "disconnected terrain" if it satisfies the following two conditions. First, there must be more than one point in the height buffer. Second, if those points are found, the standard deviation of the heights of those points must be less than $\delta_{\min}/2$, which prove those points are located over a flat terrain surface. If the two conditions mentioned above are satisfied over ϕ, the ϕ is proven as the "disconnected terrain", and the UTF is triggered. With the increased height buffer, most points located over the court yard are populated as on-

terrain candidate points, one of which will be labelled as on-terrain point by the UTF. In this way, the entire surface of the court yard can be classified as the terrain surface.

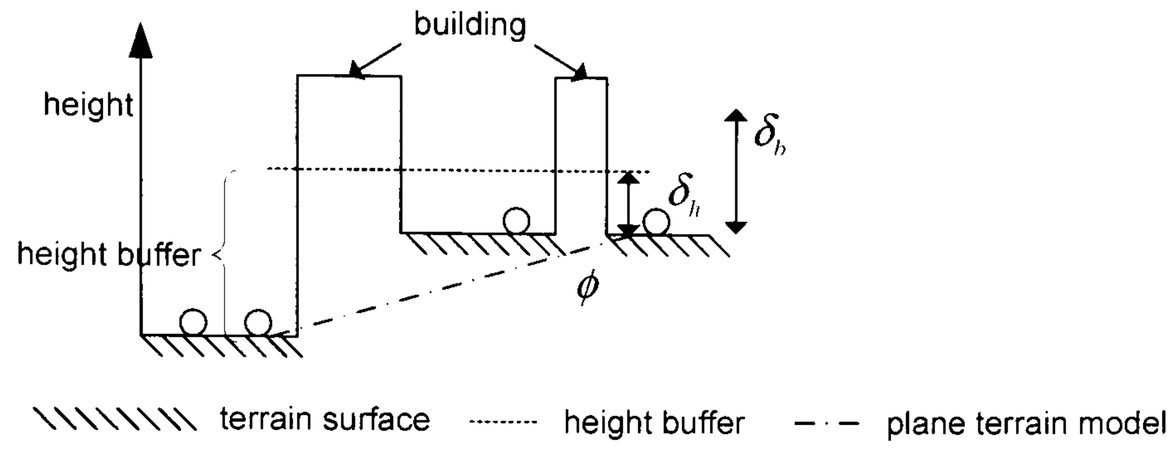


Figure 4-17. Height buffer generation for the "disconnected" terrain

The problem is, however, that a bridge object may be also classified as the "disconnected terrain" by the method mentioned above even though the bridge object must be classified as off-terrain feature. The bridge usually has a large slope discontinuity from the road or river underneath it, while the bridge is also connected to the terrain surface in almost the same height. As a result, a terrain model of ϕ, which is constructed at the end of the bridge, would be classified as the "disconnected terrain" and the UTF is triggered over the bridge surface (see ϕ_1 and ϕ_3 in figure 4-18). Thus, the bridge objects must be pre-classified before the "disconnected terrain" classification is made over ϕ.

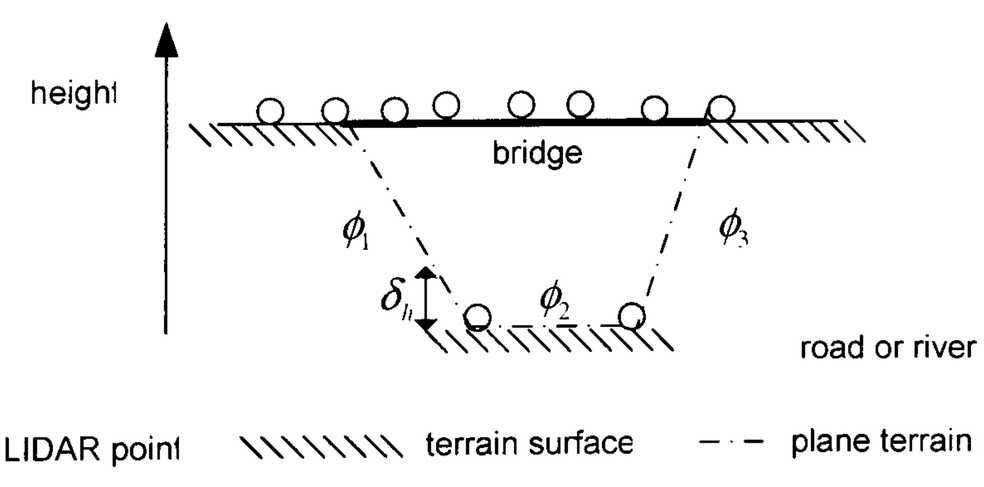


Figure 4-18. Classification of the "bridge terrain"

To this end, a "bridge terrain" class is defined. It is assumed that the bridge objects lies over a plane surface such as rivers or roads with certain height discontinuity and the surface of the bridges are almost flat. Based on this, a "discontinuous terrain" is classified as the "bridge terrain" if relative heights between the model and bridge points

are larger than δ_h, and the standard deviation of the relative heights is less than $\delta_{\min}/2$. The ϕ_2 in figure 4-18 shows an example of the "bridge terrain". Once ϕ_2 is classified as the "bridge terrain", the member points of ϕ_2 are labelled as off-terrain points, and these labels are unchanged in remaining process of UTF. This process prevents the UTF from triggering over the bridge surface. In order to make the "bridge terrain" classification sensible, the "disconnected terrain" classification is made after as many points as possible are examined by the "bridge terrain" condition. To this end, the UTF is performed in a scaleable model space. At the first iteration, UTF is performed over a small number of terrain models with relatively low slope until no on-terrain point is detected. In the next iteration, more terrain models with higher slope are involved in the UTF. This iteration is terminated when the entire model with a full range of terrain slope are used for the UTF. In this way, discontinuous terrains are reconstructed as terrain surfaces later after the "bridge terrain" condition is examined over all the nearly flat terrain surfaces. Figure 4-19 shows a functional flow of the modified UTF process mentioned above, where a terrain model, ϕ_j, is investigated with the "bridge terrain" condition, followed by the "disconnected terrain" condition.

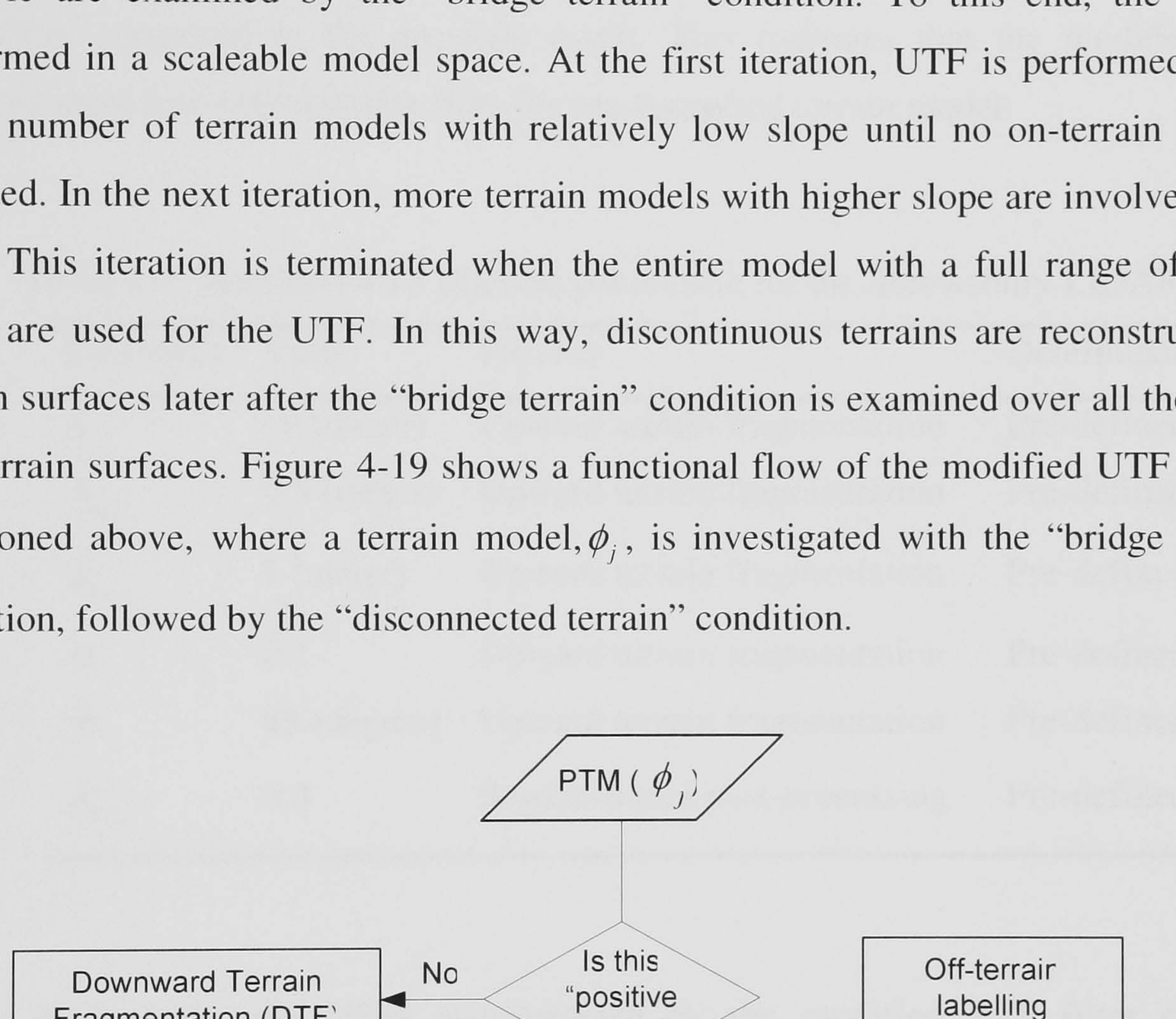


Figure 4-19. Illustration of a functional flow of the modified UTF process; gray blocks represent new procedures developed for the modified UTF process

The modified RTF filter was applied to the Shrewsbury data. Table 4-7 summarizes parameters used for the DTM generation. Three parameters are added to the previous ones for the modified RTF filter. For defining a horizontal extent of low objects, κ was selected as 10 metres. $\delta_{\min}$ and $\delta_{\max}$ were chosen as 0.5 and 1 metres respectively. With a total of 503,212 LIDAR points, 440,083 points are detected as on-terrain points, while 63,129 points are detected as off-terrain points. This result shows that the on-terrain points decrease by 1.2 %, while the off-terrain points increases by 9.4% by the modified RTF filter compared to the previous result. This indicates that the modified filter removed more low object points from the reconstructed terrain model.

Table 4-7. Modified RTF filtering parameters for the Shrewsbury LIDAR DSM

Parameter	Value	Process	Determination
κ	10 (meter)	Upward terrain fragmentation	Pre-defined
$\delta_{\min}$	0.5 (meter)	Upward terrain fragmentation	Pre-defined
$\delta_{\max}$	1 (meter)	Upward terrain fragmentation	Pre-defined
α	0.1	Upward terrain fragmentation	Pre-defined
β	45 (degree)	Upward terrain fragmentation	Pre-defined
δ_{PC}	0.3	Region-based post-processing	Pre-defined

Figure 4-20 shows the DTM reconstructed by the modified RTF filter over the Shrewsbury data. The result shows that the modified filter obviously makes the terrain surface much smoother, while the overall terrain shapes are well preserved. In addition, some bridges are successfully removed as off-terrain objects. It suggests that the adaptive strategy to use different value of δ_h depending on the size of terrain models works efficiently.

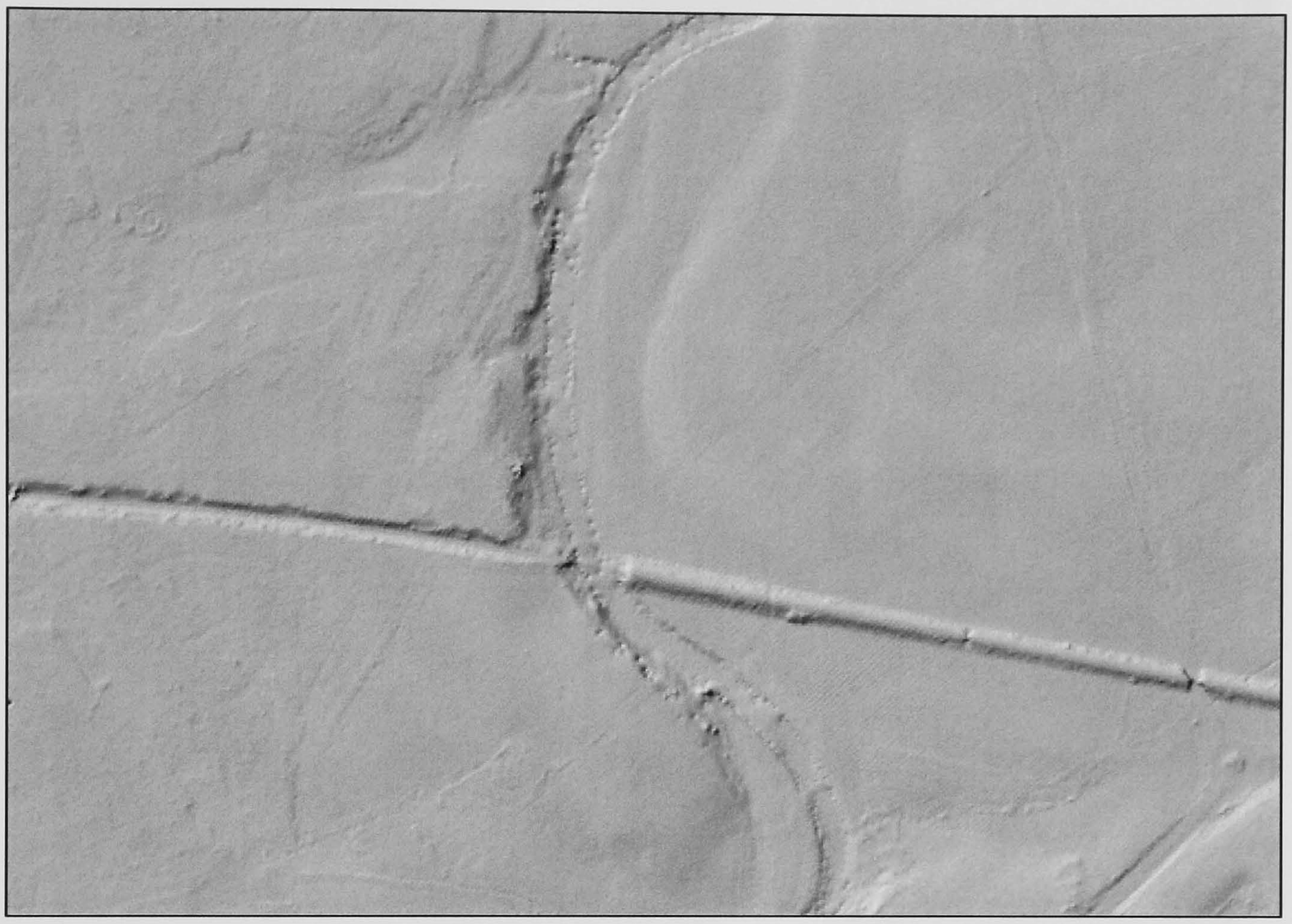
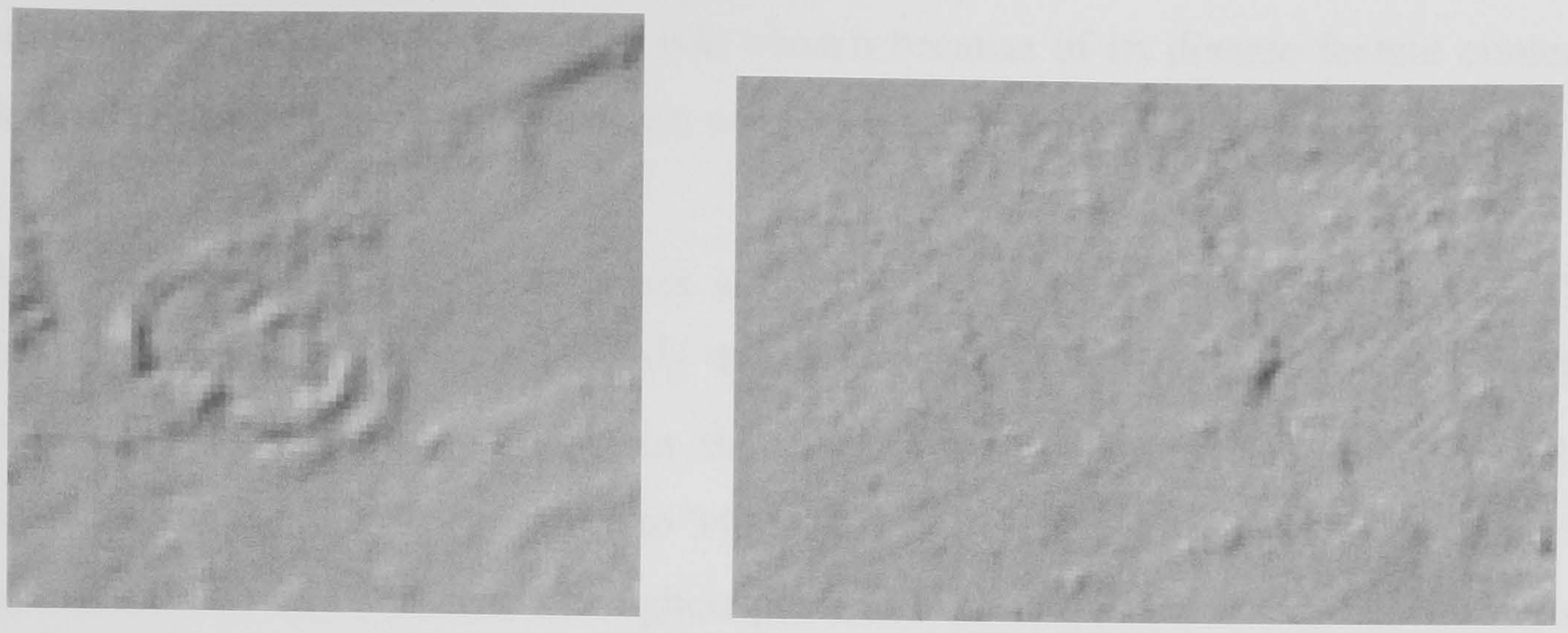


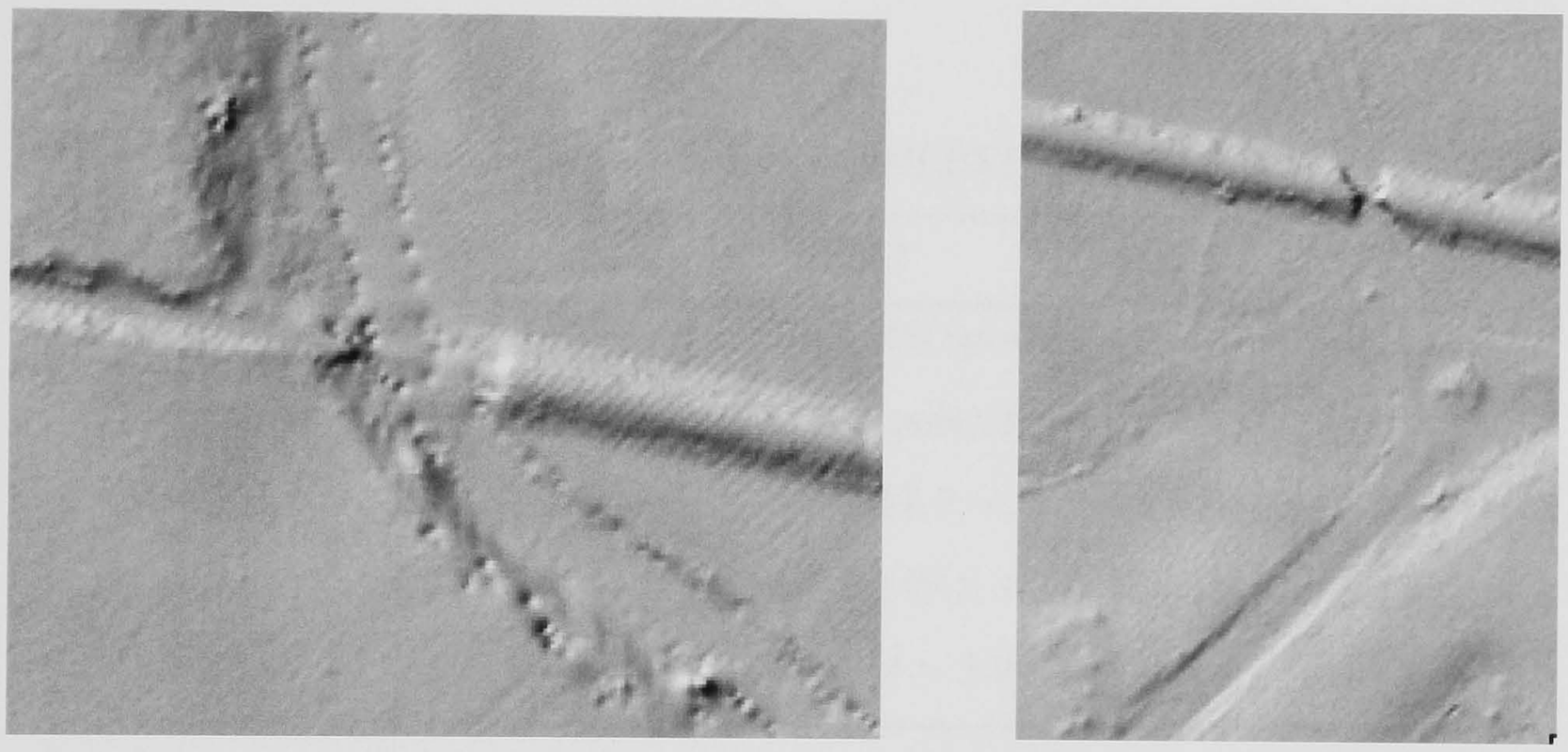
Figure 4-20. Modified RTF filtering result over the Shrewsbury LIDAR DSM

Figure 4-21 shows the results of the modified RTF filter over four sub-scenes which were used in §4.2.4. Over residential areas such as Area-1 and Area-2, more low objects are removed, but the geomorphic shape of the knoll in Area-1 is well preserved (see figure 4-21 (a) and (b)). The railway bridges shown in figure 4-21 (c) and (d) are taken as off-terrain objects, which shows that those area were recognized as the "bridge terrain" class, not as the "disconnected terrain" class, even though nearby railway embankment has the similar height to the bridge. In particular, the hedges in the agricultural field and along the road are well removed. It implies that $\delta_{\min}$ of 0.5 m is suitable for removing these objects. However, very low objects in Area-2 still remain as the terrain surface. If the RTF filter employs an interpolation method during the process, the terrain surface can be artificially smoothed, but this depends on the application required.



(a) Area-1

(b) Area-2



(c) Area-3

(d) Area-4

Figure 4-21. Modified RTF filtering results of four sub-areas selected in the Shrewsbury LIDAR DSM

4.4 CASE II: OEEPE DATA

4.4.1 OEEPE LIDAR DSM

The modified RTF filter described in the previous section was applied in an urban environment. As part of the second phase of the OEEPE project on laser scanning (OEEPE, 2000), LIDAR DSMs were acquired over the Vahingen/Enz test field and Stuttgart city center by an Optech ALTM scanner. Both first and last pulse data were recorded. The Working Group III/3 of ISPRS Commission III selected eight sites from the OEEPE dataset for the comparison of LIDAR filtering algorithms, which were developed by international institutes and companies (Sithole & Vosselman, 2003b). Amongst them, an urban area was selected from the OEEPE LIDAR DSMs for

validating the RTF filter. This area was chosen because of its diverse feature content. Only last pulses of the OEEPE data set were used for the current experiment.

The OEEPE LIDAR DSM comprises a total of 243,403 LIDAR measurements. The point density is about 0.91 (points/m^2). The dimensions of the test data are 421 m in the East-West direction, and 631 m in the North-South direction. In the study area, elevation values range from 248 m to 348 m. The lowest terrain points can be found in the North-West corner, while the highest points are located in the South-East corner of the study area. Table 4-8 summarizes the data characteristics of the OEEPE LIDAR DSM.

Table 4-8. Data characteristics of the OEEPE LIDAR DSM

Parameters	Values
Total number of LIDAR points	243,403 (points)
Point spacing	0.91 (point/meter^2)
X range	513449.5 ~ 513870.5 (metre)
Y range	5402649.5 ~ 5403280.5 (metre)
Z range	248.41 ~ 348.29 (meter)

Figure 4-22 and 4-23 show the OEEPE LIDAR DSM. It shows a range of the object variety and complexity, which could make the filter fail to properly classify LIDAR points. The sizes and shapes of buildings in the study area vary a lot, and range from a simple flat roof building to a very complex façade. Those buildings are built up over a flat terrain in the Western part of the study area, while over a sloping terrain in the Eastern part. There are many bridges with different sizes. A narrow bridge can be seen in the North-West, and there is a wider bridge in the South-West. Also, some small bridges can be shown in the North-East and the South-West. There is a range of terrain types in the study area. Due to some construction work most of terrains located close to the buildings and roads have sharp ridges, which is similar to building boundaries. In particular, terrain segments are not smoothly connected to neighbouring terrain surface in the Eastern part of the study area. Some courtyards which are encompassed by buildings can be found, and these are disconnected from nearby terrains though their heights are similar. There is also a ramp in the North-East, which is sloping from a

building site to the ground. Vegetation objects are scattered in the entire study area and most of them are very low objects. Thus, the OEEPE LIDAR DSM can be considered as a good test site to validate how the RTF filtering rationale can work in such a complex urban environment.

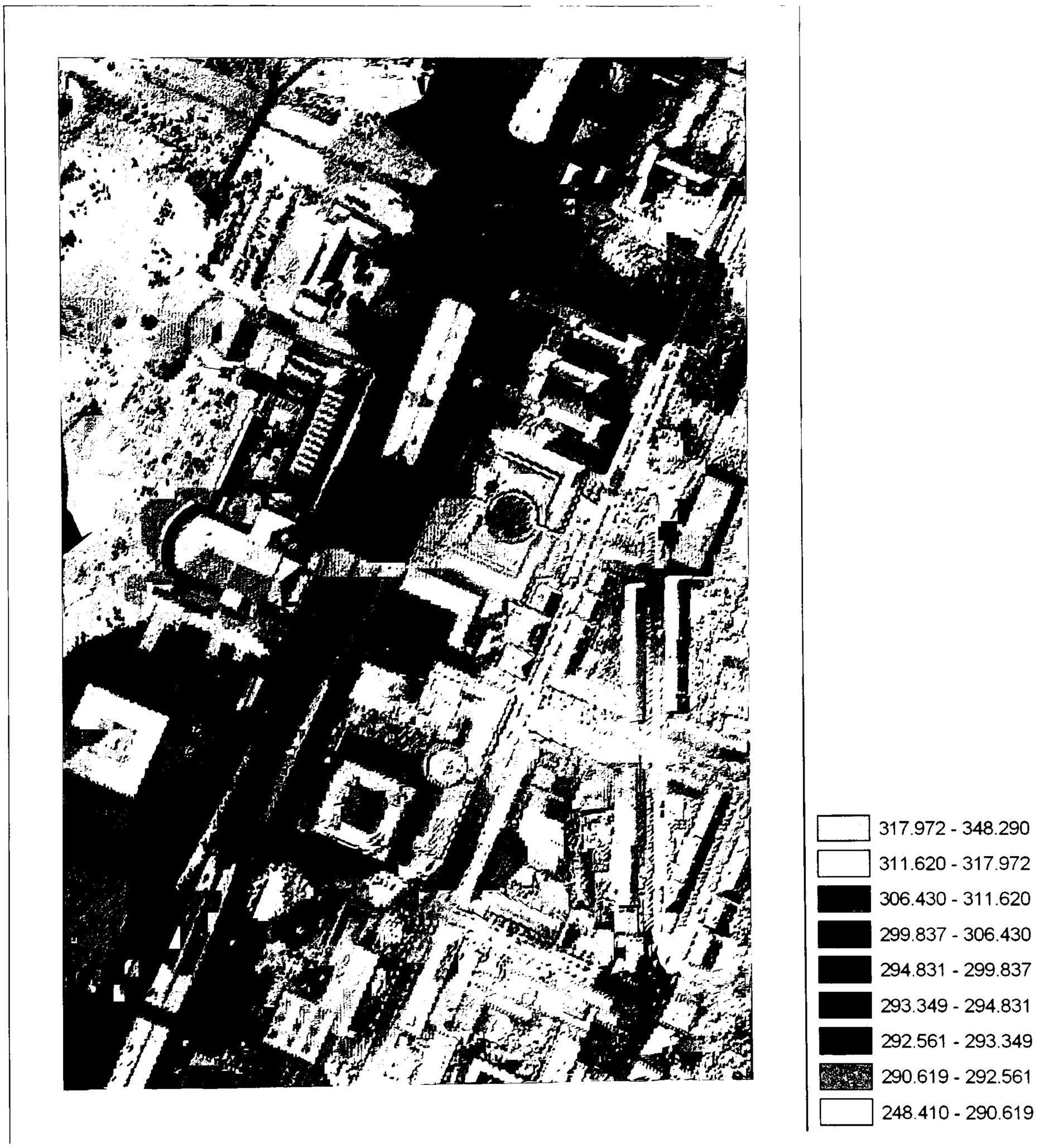


Figure 4-22. OEEPE LIDAR DSM

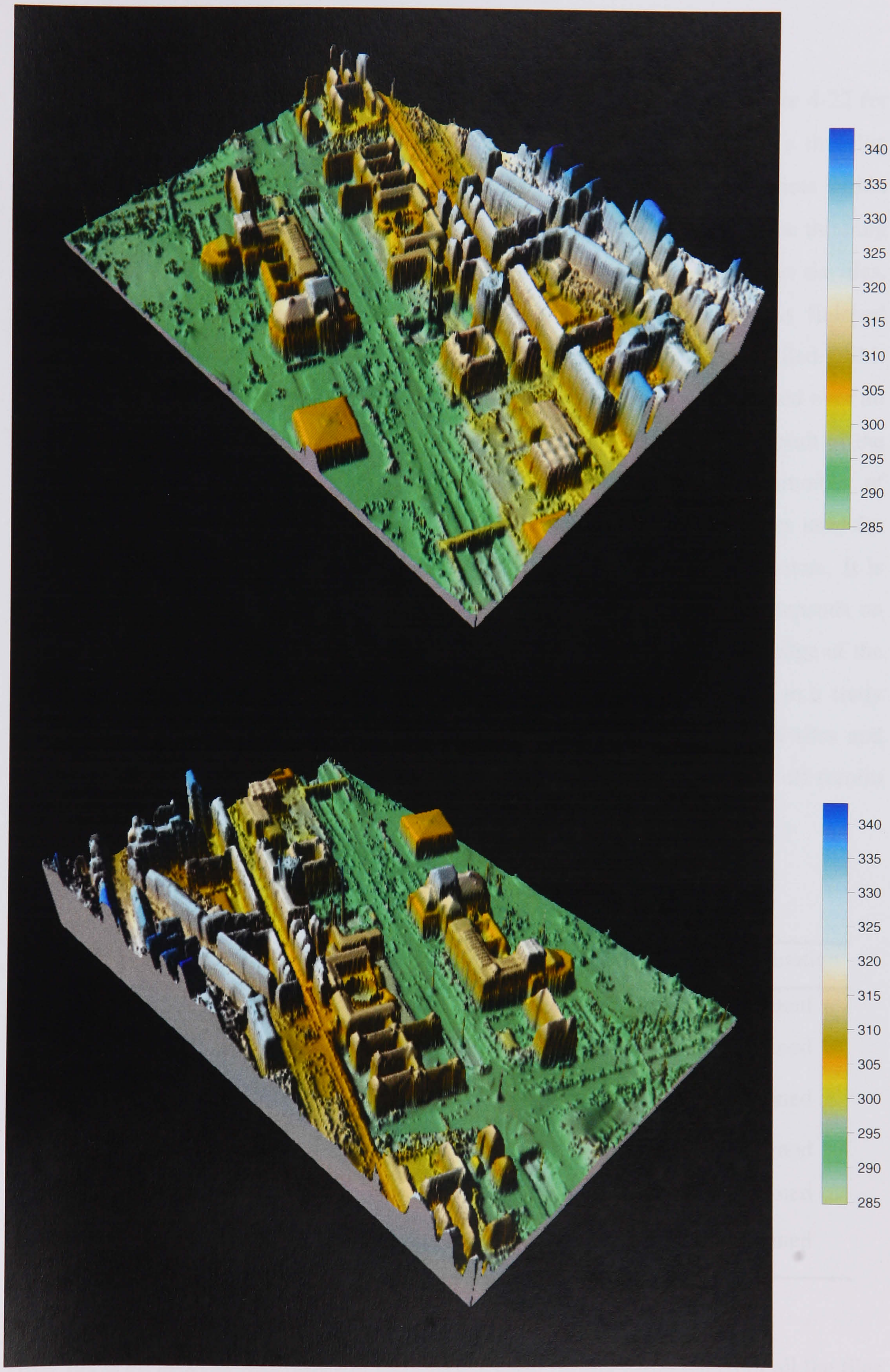


Figure 4-23. OEEPE LIDAR DSM with different perspective views

4.4.2 Automatic DTM generation result

The RTF filtering algorithm was applied to the OEEPE LIDAR DSM of figure 4-22 for the generation of the DTM. Similar to the case of the Shrewsbury data, a thinning process removed those points which are closely spaced with neighbouring points within 1 cm, and a total of 243,403 LIDAR points were reduced to 220,508. With the thinning procedure, a tiling method automatically divided the entire LIDAR DSM into six tiles, each of which covers 0.21 km x 0.21 km. Once the data preparation was finished through the thinning and tiling process, the RTF filter was applied to each tiled DSM. After initial terrain models were generated for six tiles, the DTF process started over all the initial terrain models and detected a total of 529 on-terrain points. The result of the DTF provides a rough terrain model where the UTF is triggered. A number of parameters were pre-determined for the UTF, which are the same as the ones used for the Shrewsbury data described in §4.3. Table 4-9 summarizes those parameters. It is noteworthy that the selection of the parameters used for the UTF does not depends on the resolution of LIDAR DSM, but is determined based on a heuristic knowledge of the maximum size of low objects, and the maximum variation of terrain slopes in a study area. With these parameters, the UTF process was applied to all the DSM tiles and recognizes 140,798 LIDAR points as on-terrain points, and 79,710 points as off-terrain points.

Table 4-9. RTF filtering parameters for the OEEPE LIDAR DSM

Parameter	Value	Process	Determination
κ	10 (meter)	Upward terrain fragmentation	Pre-defined
$\delta_{\min}$	0.5 (meter)	Upward terrain fragmentation	Pre-defined
$\delta_{\max}$	1 (meter)	Upward terrain fragmentation	Pre-defined
α	0.1	Upward terrain fragmentation	Pre-defined
β	45 (degree)	Upward terrain fragmentation	Pre-defined
δ_{PC}	0.3	Region-based post-processing	Pre-defined

After the UTF process was finished, the classification results extracted from all the tiles were integrated as one dataset, over which the post-processing was applied. With δ_{PC} of

0.3, the region-based post-processing converted 158 on-terrain points to off-terrain ones. Then, the point-based post-processing corrected 90 off-terrain points to on-terrain points. Finally, the RTF filter detected 140,730 on-terrain points and 79,778 off-terrain points. Table 4-10 summarizes the RTF filtering results of the OEEPE LIDAR DSM.

Table 4-10. RTF filtering results over the OEEPE LIDAR DSM

Processing step	Results (points)	
	on-terrain	off-terrain
Downward terrain fragmentation (DTF)	529	219979
Upward terrain fragmentation (UTF)	140798	79710
Region-based post-processing	140640	79868
Point-based post-processing	140730	79778
Total		220508

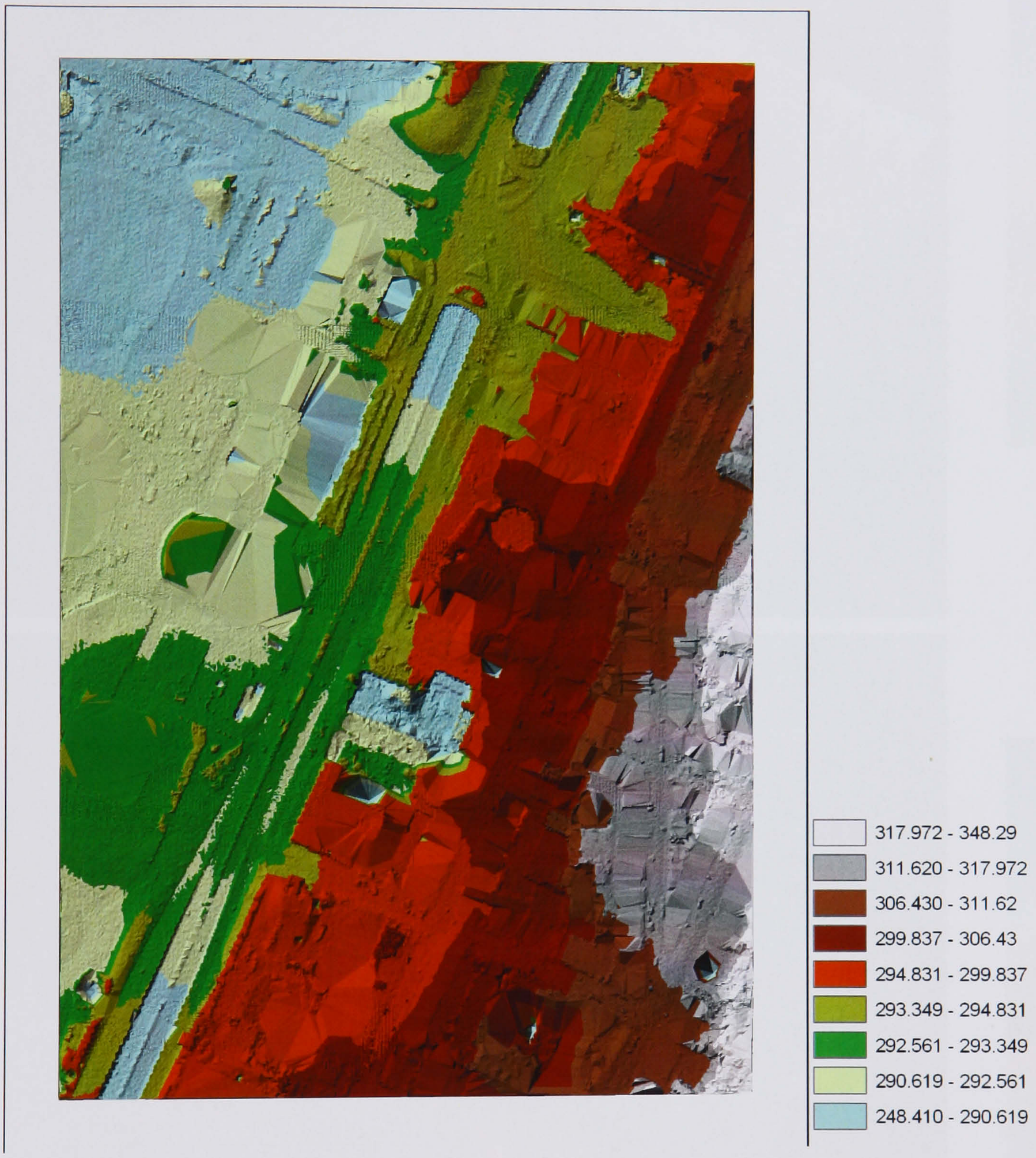


Figure 4-24. RTF filtering result in 2D over the OEEPE LIDAR DSM

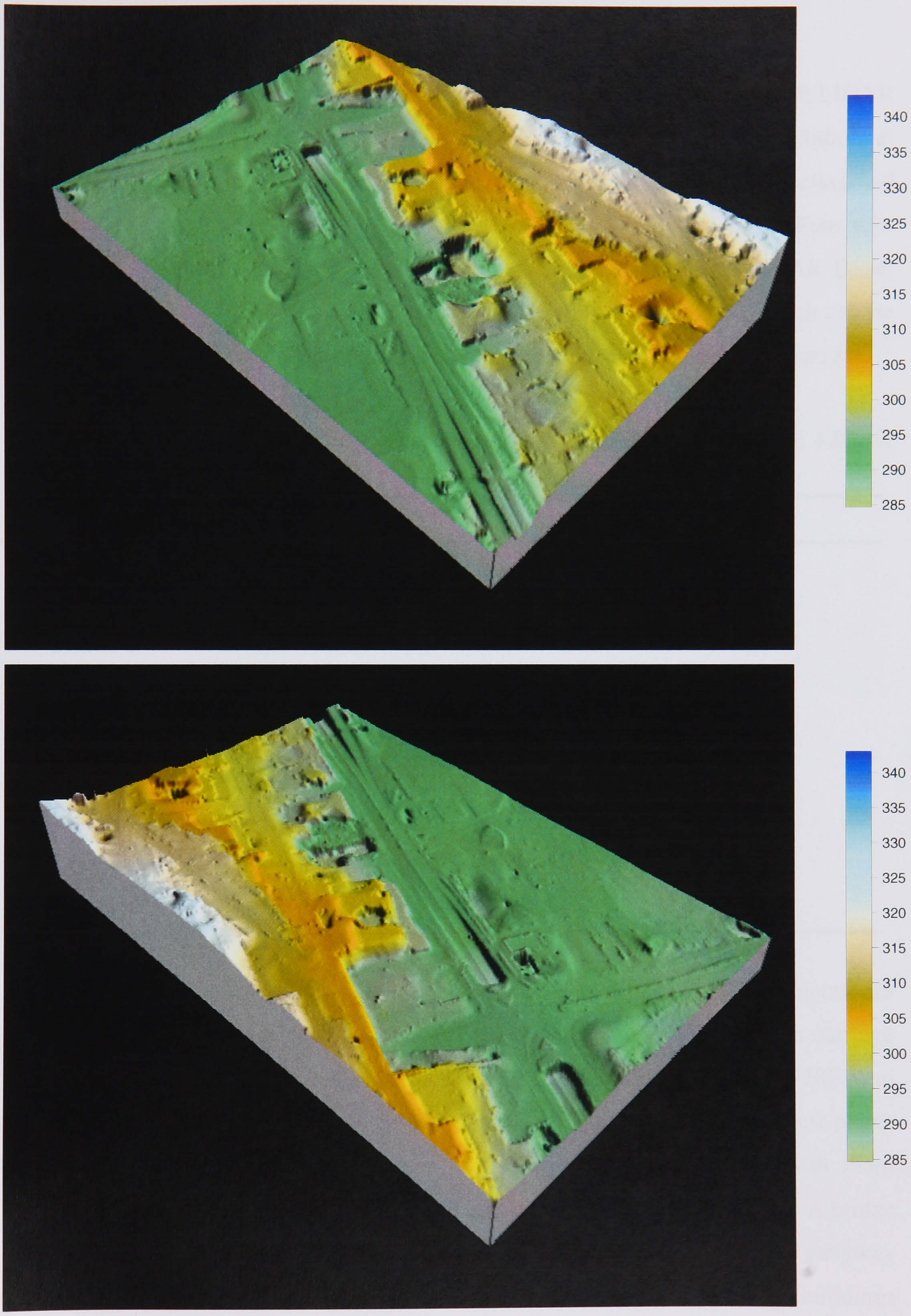


Figure 4-25. RTF filtering result in 3D with different perspective views over the OEEPE LIDAR DSM

4.4.3 Quality assessment

The RTF filtering algorithm was examined as the filtering result of the OEEPE LIDAR DSM was compared to a reference data. The reference data was generated by Sithole & Vosselman (2003b), in which on-terrain and off-terrain points were manually classified with the knowledge of landscape in the study area and some aerial images. Four sub-scenes were chosen for producing the reference data from the OEEPE LIDAR DSM, which are shown in figure 4-26. Each sub-scene has interesting features, which can be used for validating the RTF filtering algorithm. Table 4-11 summarizes these features.

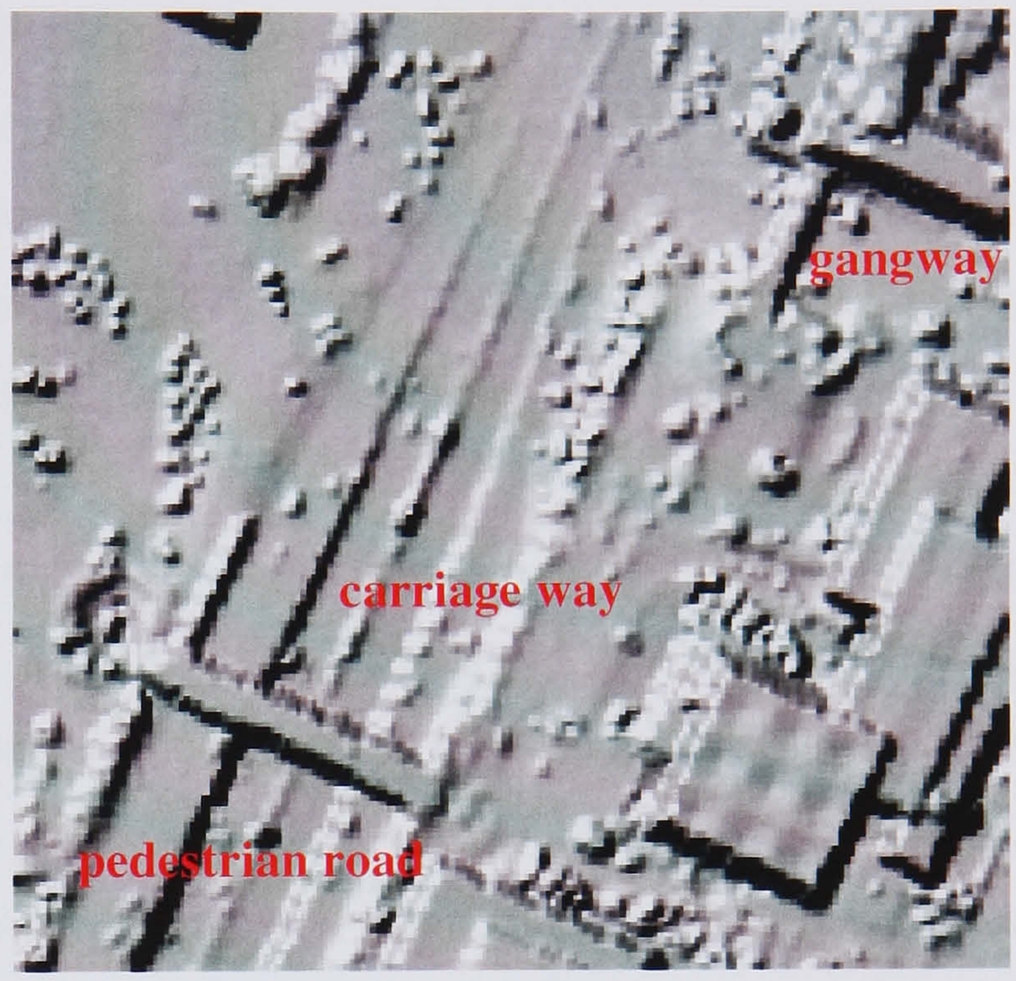
Table 4-11. Characteristics of four sub-scenes selected from the OEEPE LIDAR DSM

Sub-scene	Interesting features
Area-1	Narrow bridge
	Low objects
Area-2	Bridge (South-West)
	Gangway (North-West)
	Discontinuous terrain
	Low objects
Area-3	Complex buildings
	Disconnected terrain
Area-4	Low objects on the ramp terrain

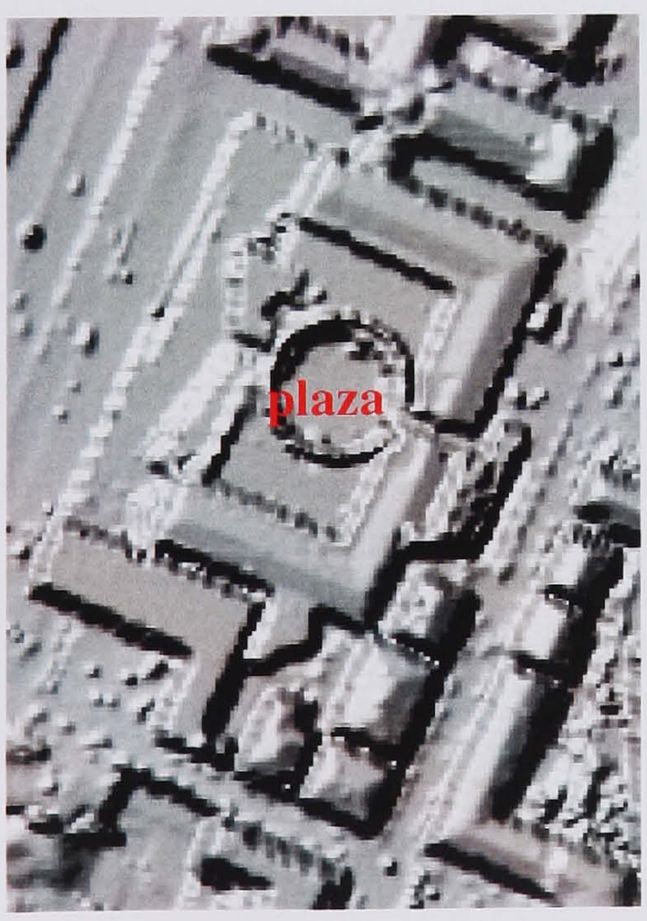
Area-1 shows flat terrain where a narrow bridge is elevated and low objects are scattered. A discontinuous terrain is a dominant feature in Area-2. Around a carriage way, there is a staircase-like terrain surface, where the elevation ranges from 287 m to 298 m. The lowest terrain point can be found in the carriageway, and the highest points are located nearby large buildings in the Eastern part. A pedestrian road with sharp ridge is located close to the carriageway with a height difference of about 6 m. A bridge can be found in the South-West which is elevated from the carriage way and a gangway in the North-West is located between a building and terrain. Area-3 has large buildings with very complex shapes. Three sides of a plaza in the centre of the scene is surrounded by a block of buildings. From the plaza, it is possible to walk onto the road to the East and also descend through stairs to the road in the West. Moreover, there is a

sunken arcade in the center of the plaza. In Area-4, there is a ramp terrain, which is sloping from a building site to the terrain, and many low objects are located on the ramp terrain.

(a) Area-1



(b) Area-2



(c) Area-3

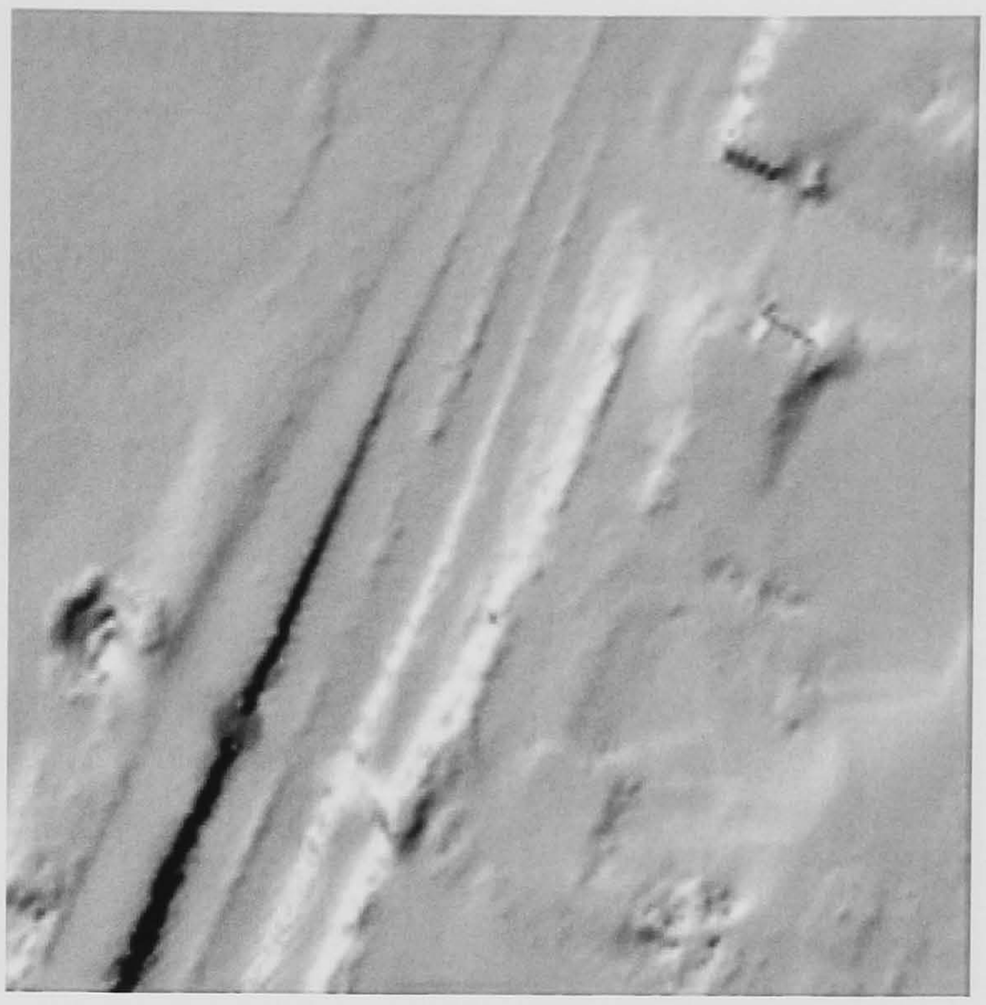
(d) Area-4

Figure 4-26. LIDAR DSM of four sub-scenes selected from the OEEPE LIDAR DSM

Over the four sub-scenes, reference DTMs and filtered DTMs were generated with manually classified on-terrain points and the RTF filtering result respectively. Figure 4-27 and 4-28 show the reference DTMs and filtered DTMs respectively. From a quick visual inspection of the results, it can be concluded that the RTF filter successfully preserves the overall geomorphic shape of underlying terrain surfaces, and significant terrain segments are not missed in the DTM reconstructed by the filter. This implies that the RTF filer does not suffer from difficulties in detecting discontinuous terrain surfaces. Moreover, most buildings and bridges were removed by the filter. However, the reference DTMs show smoother terrain surface than the filtered DTMs. This suggests that some low objects are recognized as mistakenly on-terrain objects by the RTF filter.

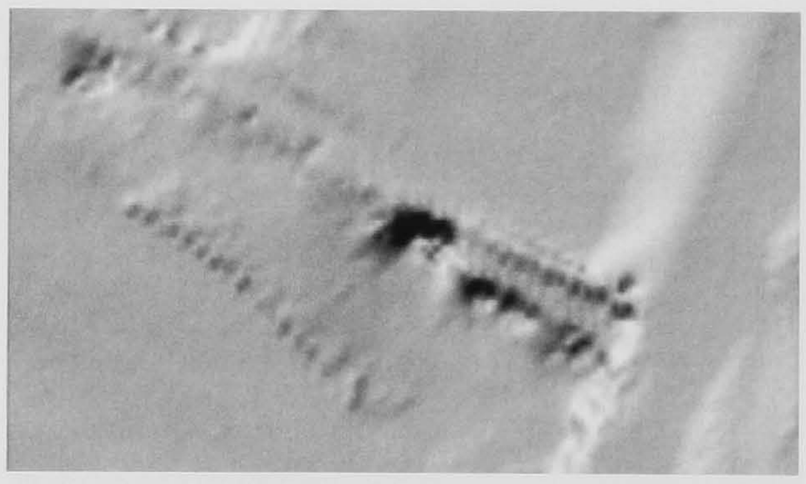
For a quantitative analysis of the RTF filter, labelling errors produced by the RTF filter were measured in comparison with the reference data. To this end, two error types (i.e., Type I and Type II error) are defined. Type I error is generated when a filter misclassifies a real on-terrain point as an off-terrain object. As Type I errors increase, more terrain segments will be missed in a terrain model reconstructed. Type II errors are a reciprocal of Type I error. Type II error occurs when a filter mistakenly recognizes a real off-terrain point as an on-terrain object. As a result, distracting off-terrain features can be involved in the construction of a terrain model. Both Type I and Type II error are related to specific features in landscape. Type I errors are produced when a terrain surface is similar to the shape of off-terrain object or a terrain segment is completely surrounded by off-terrain objects. A discontinuous terrain with sharp ridges and a disconnected terrain such as a courtyard are good examples of the former and the latter case. Type II errors will occur if the surface of an off-terrain object is smooth enough to be compatible with terrain surfaces. A very low object may cause the Type II error since its height difference from underlying terrain is not distinguishable. Therefore, Type I and Type II errors can be used to analyze the relationship of a filtering algorithm to features in the landscape.

(a) Area-1



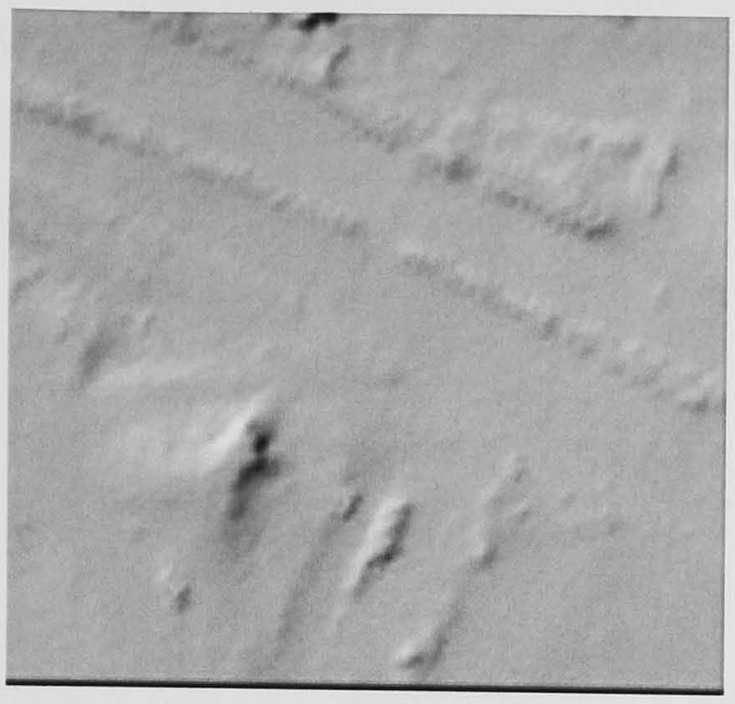
(b) Area-2

(c) Area-3

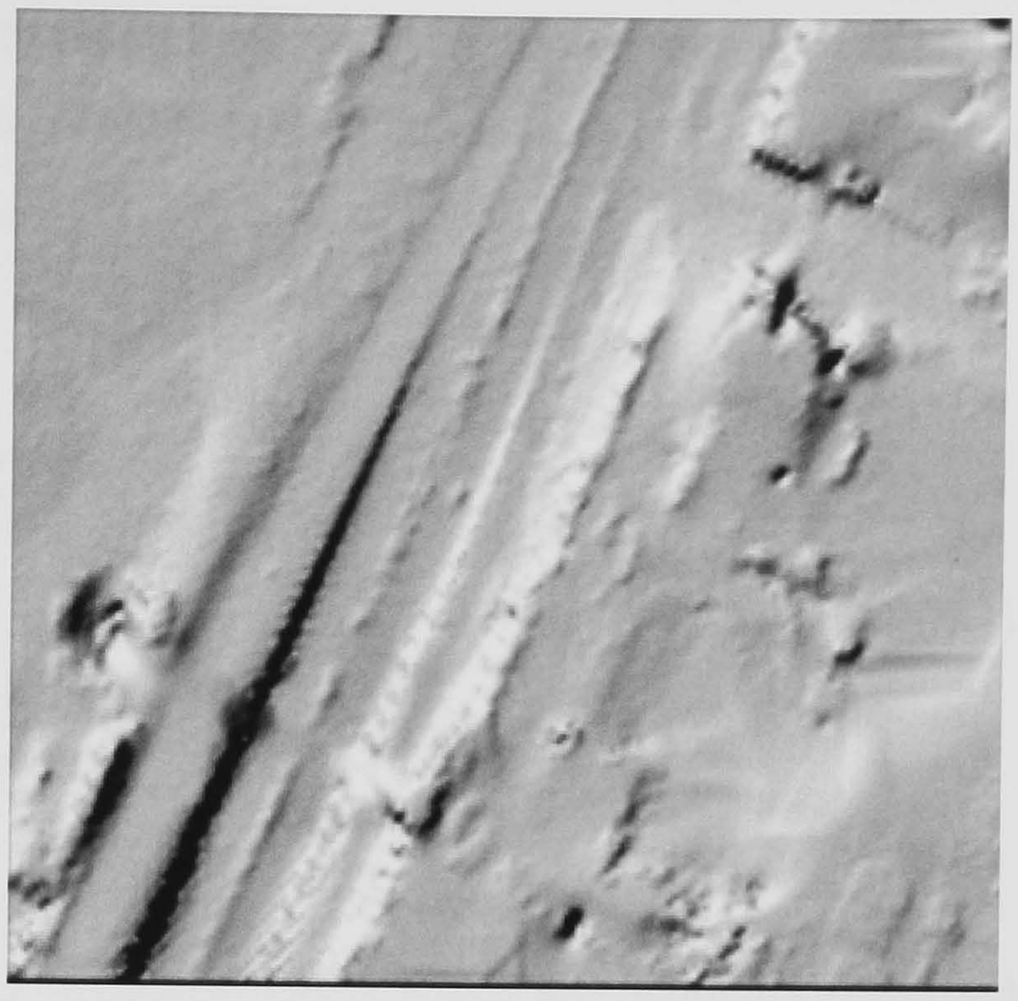


(d) Area-4

Figure 4-27. Reference DTMs of four sub-scenes selected from the OEEPE LIDAR DSM



(a) Area-1



(b) Area-2

(c) Area-3

(d) Area-4

Figure 4-28. Reconstructed DTMs by RTF filter over four sub-scenes selected from the OEEPE LIDAR DSM

For evaluating Type I and Type II errors, manually classified labels of the reference points were point-wisely compared to the RTF filtering result. Type I and Type II errors are then quantified depending on a ratio of the mismatched labels over the total number of each label as follows:

$$Type\ I\,(\%) = \frac{Total\ number\ of\ on\text{-}terrain_{MIS}}{Total\ number\ of\ on\text{-}terrain_{REF}} \times 100$$

$$Type\ II\,(\%) = \frac{Total\ number\ of\ off\text{-}terrain_{MIS}}{Total\ number\ of\ off\text{-}terrain_{REF}} \times 100 \qquad (4\text{-}4)$$

where $on\text{-}terrain_{REF}$ and $off\text{-}terrain_{REF}$ represent labels which are manually classified as on-terrain and off-terrain point respectively; $on\text{-}terrain_{MIS}$ is a label which is recognized as an on-terrain point by the manual classification, but as an off-terrain point by the RTF filter; $off\text{-}terrain_{MIS}$ represents a label misclassified by the RTF filter for a reference off-terrain point.

Over the four sub-scenes, the RTF filter detected 44,141 on-terrain points and 22,273 off-terrain points. Once this result is compared to the reference data, 0.94% labelling error was computed as Type I error in which 417 points of 44,558 reference on-terrain points are misclassified as off-terrain points, while 6.8% for Type II errors that misclassifies 1,611 of 23,884 reference off-terrain points as on-terrain points. This result can be seen in a cross-matrix of table 4-12.

Table 4-12. Cross-matrix of a total RTF filtering error computed from four sub-scenes of the OEEPE LIDAR DSM

Total		Filtered		Total	Error		Min	Max	Mean	Std Dev.
		on	off	points	Type	%	(m)	(m)	(m)	(m)
Reference	on	44141	417	44558	I	0.936	-1.11	4.20	0.50	0.628
	off	1611	22273	23884	II	6.745	-38.4	12.83	0.88	1.970

In table 4-12, Type I error is much lower than Type II error. As mentioned earlier, smaller Type I errors means that the RTF filter is better adaptive to the terrain discontinuity, whereas larger Type II errors means that the filter has a drawback not to completely remove low objects. This conclusion agrees to the result of a visual comparison between the filtered DTMs and the reference DTMs, which was described in §4.4.3.

In addition to the ratio analysis, a statistical analysis for both errors was performed. Error points of Type I and Type II were projected onto the reference DTMs of figure 4-27, from which height differences between the points and the reference DTMs were measured. Most of the RTF filtering errors are occurred with the points whose heights are very low from the reference DTMs. The mean values of the height difference between the Type I error points and the reference DTMs is 0.5 m, while 0.88 m for the Type II error. From this result, it can be concluded that the RTF filtering errors does not significantly degrade the overall quality of the terrain model reconstructed, and errors produced by the filter are only limited to very low objects less than 1 m in height from the terrain surface. In addition, this result suggests that Type I errors can be easily removed by a simple thresholding. For instance, Type I error can be greatly reduced by re-labelling off-terrain points produced by the filter as on-terrain points if their height difference from the reconstructed terrain surface is less than 0.5 m. However, Type II errors are rather difficult to be corrected since they are already involved in the terrain model reconstructed.

Table 4-13 shows the cross-matrices computed from all the sub-scenes. In table 4-10, area-1 shows the lowest Type I error (0.10 %), while the highest value of Type I error (1.81 %) can be found in Area-4. All the mean height differences between Type I errors and the reference DTMs are less than 0.5 m. As for Type II error, Area-1 (6.96 %) and Area-3 (5.09 %) have lower values than Area-2 (8.21 %) and Area-4 (8.35 %). All the mean height differences between Type II errors and the reference DTMs fall into a height range from 0.8 m to 1.2 m except Area-4, which shows only 0.34 m. From the result of table 4-13, however, it is rather difficult to analyze these errors in the relation of features in each sub-scene. Thus, both errors were visually represented with DSMs in figure 4-29.

Table 4-13. Cross-matrices of RTF filtering errors computed from each sub-scene of the OEEPE LIDAR DSM

Area-1

		Filtered		Total	Error		Min	Max	Mean	Std Dev.
		on	off	points	Type	%	(m)	(m)	(m)	(m)
Reference	on	8907	9	8916	I	0.10	0.24	0.49	0.33	0.085
	off	181	2421	2602	II	6.96	-0.11	3.17	0.85	0.784

Area-2

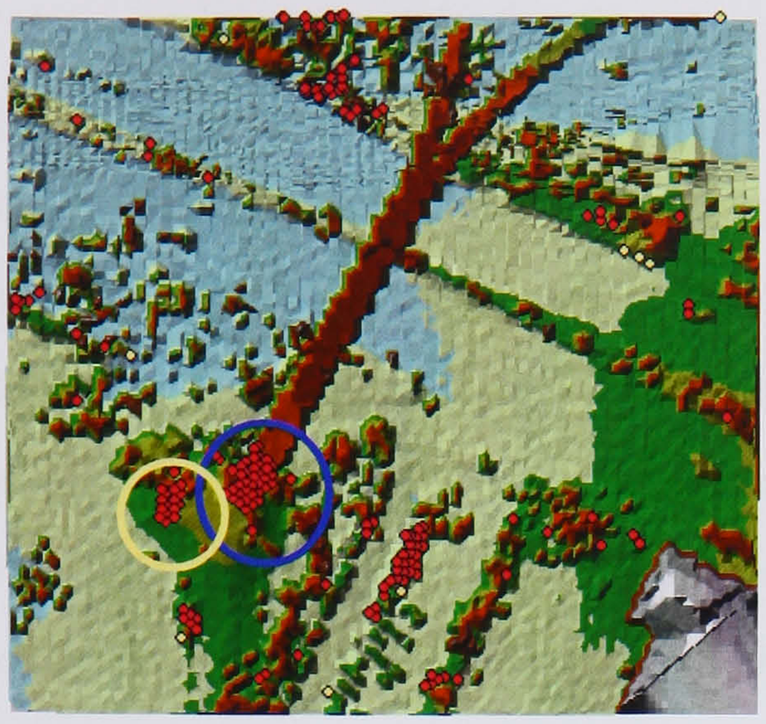
		Filtered		Total	Error		Min	Max	Mean	Std Dev.
		on	off	points	Type	%	(m)	(m)	(m)	(m)
Reference	on	20173	149	20322	I	0.73	-0.44	2.95	0.46	0.532
	off	749	8380	9129	II	8.21	-15.8	6.22	0.82	1.498

Area-3

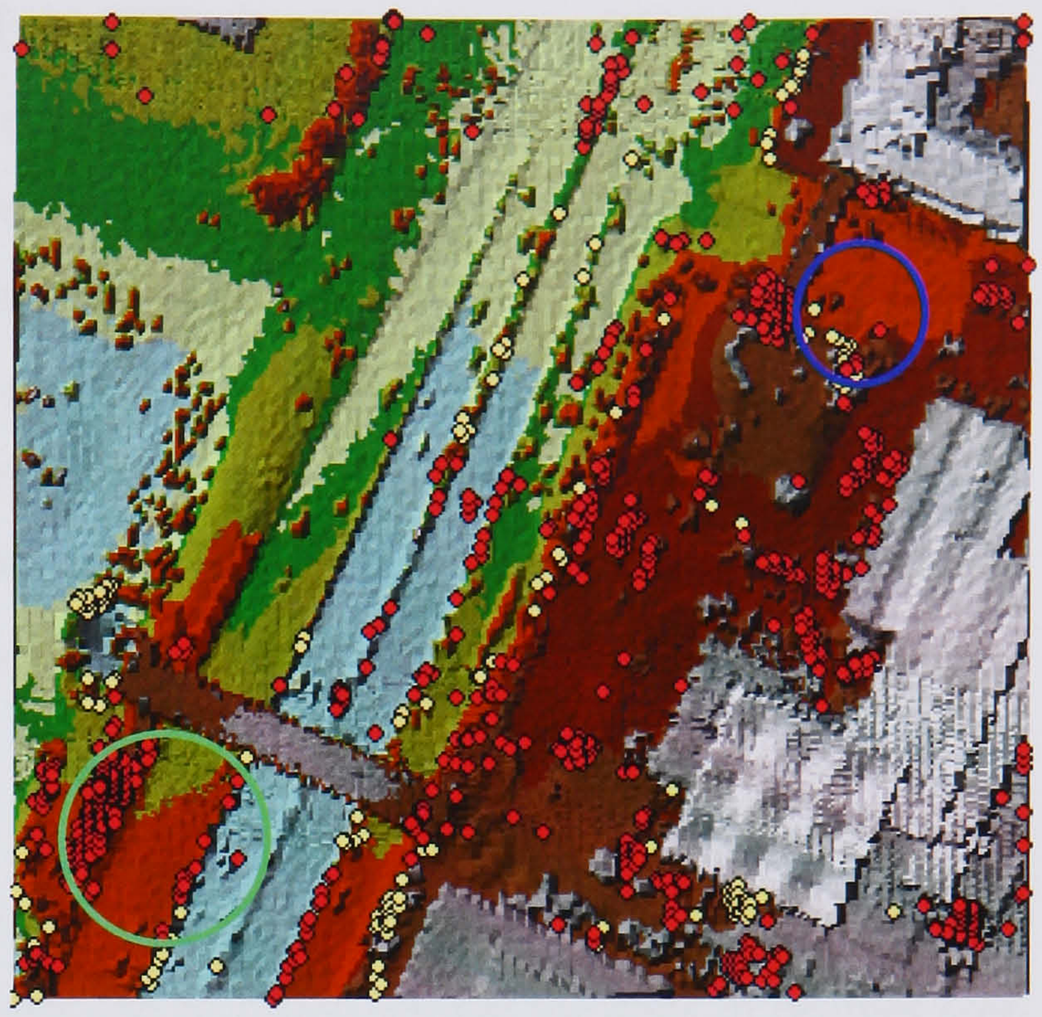
		Filtered		Total	Error		Min	Max	Mean	Std Dev.
		on	off	points	Type	%	(m)	(m)	(m)	(m)
Reference	on	10245	189	10434	I	1.81	-0.85	4.20	0.55	0.729
	off	520	9705	10225	II	5.09	-38.4	12.83	1.18	2.928

Area-4

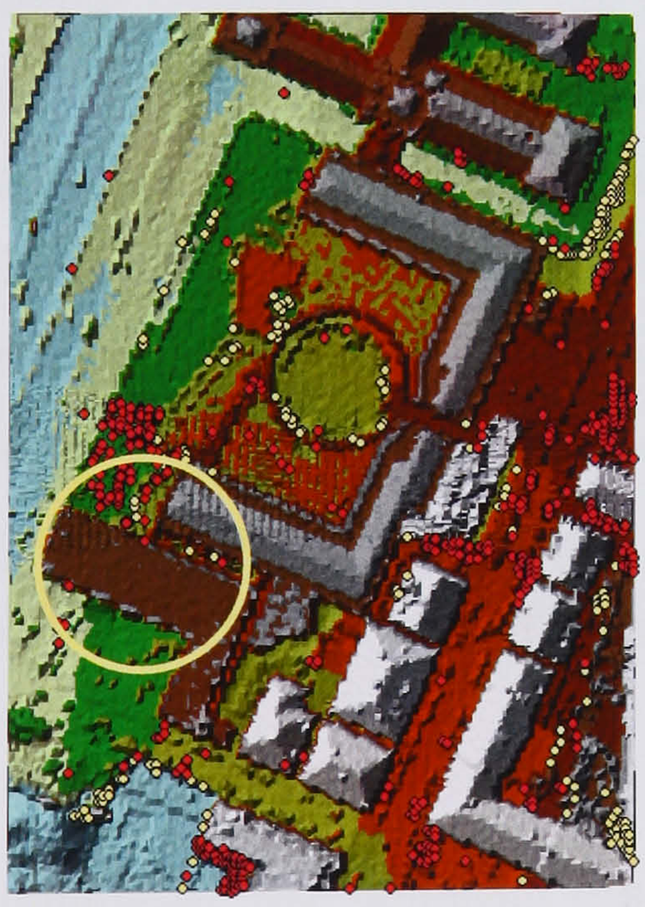
		Filtered		Total	Error		Min	Max	Mean	Std Dev.
		on	off	points	Type	%	(m)	(m)	(m)	(m)
Reference	on	4816	70	4886	I	1.43	-1.11	2.42	0.47	0.555
	off	161	1767	1928	II	8.35	3.67	-4.69	0.34	1.042



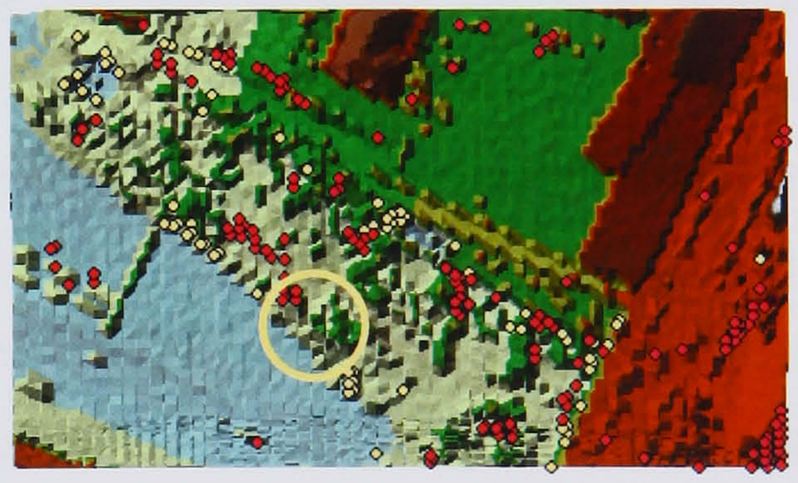
(a) Area-1



(b) Area-2



(c) Area-3



(d) Area-4

Figure 4-29. RTF filtering errors computed from four sub-scenes of the OEEPE LIDAR DSM; yellow dot represents the Type I filtering error, while red dot represents the Type II error

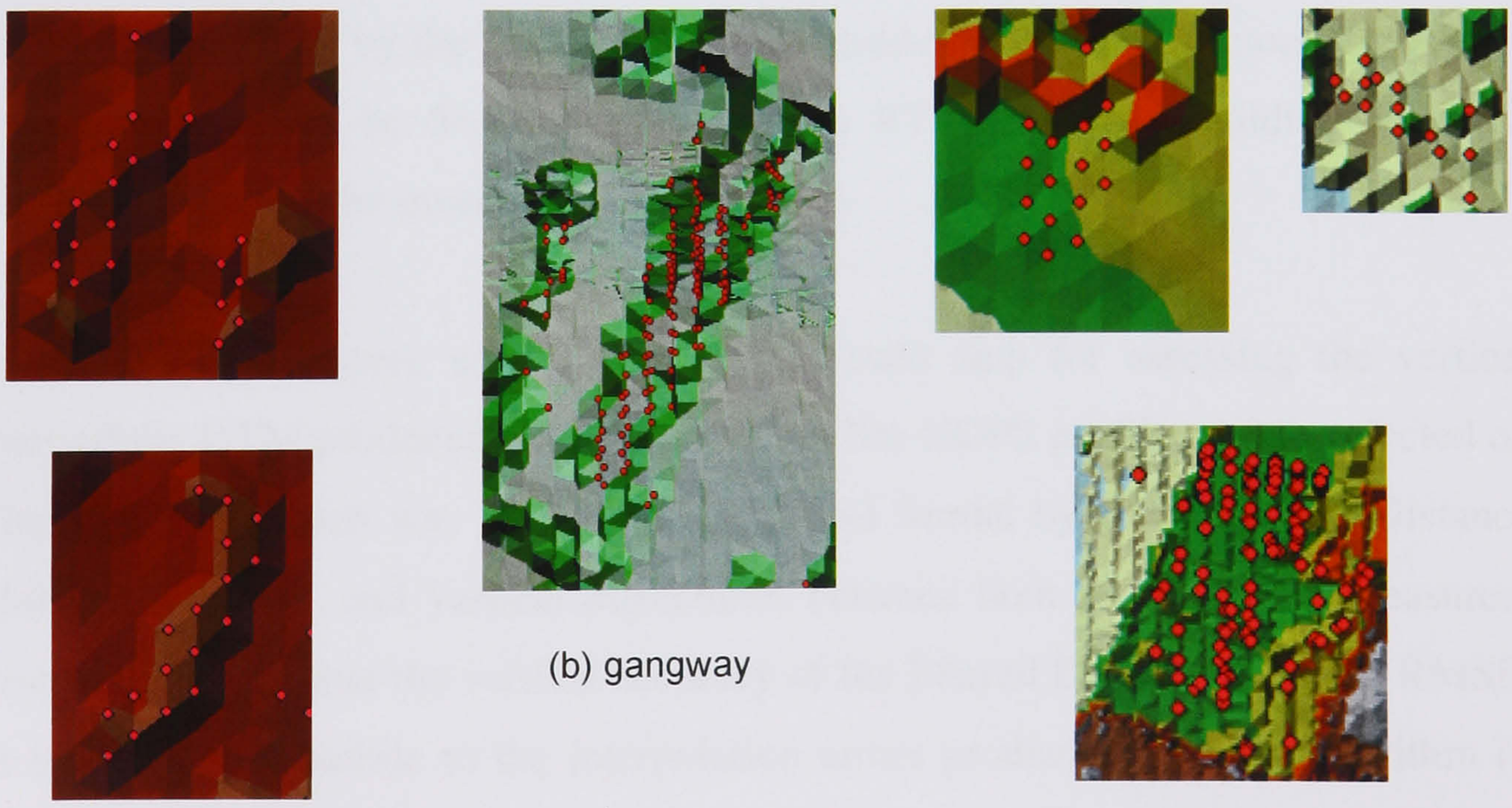
As can be seen in figure 4-29, most of Type 1 errors are closely located to discontinuous terrains and building boundaries. Thus, these errors may be generated during the "disconnected terrain" classification of the UTF. For instance, if a PTM is classified as the "discontinuous terrain", but it has only one member point, the model is rejected by the "disconnected terrain" classification since it requires more than two member points. As a result, the point located within the discontinuous terrain model is labelled as off-terrain. Although the point-based post-processing was implemented for correcting these errors, the process may fail if the error points are located on terrain surfaces where roughness is too high to post-process those points. For this reason, the smallest Type I errors can be found in Area-1 which shows the lowest terrain discontinuity than other sub-scenes.

Figure 4-29 shows Type II errors are related to two features (i.e., bridge and low objects). A large amount of Type II errors can be found around the ends of bridges (see blue circle in figure 4-29 (a) and (b)). These errors are mainly caused by the difference of filtering rationale between the RTF filter and manual classification to determine where a bridge begins and ends. The manual filtering removes bridges based on a slope similarity, where a point is removed as a bridge object if its slope is similar to the overall characteristic of the bridge though it is connected to the terrain surface at the end of the bridge. However, the RTF filter starts to recognize a bridge object when a height difference between the bridge surface and the terrain surface underneath the bridge is larger than certain threshold. As a result, the bridges manually filtered are more extended towards both ends of bridges than the RTF filtering result, which leads to the Type II errors.

Low objects are also dominant features to cause Type II errors. In particular, Type II errors produced by the RTF filter are mainly related to low objects such as small bushes which have irregular shapes in height. Some low objects which make the RTF filter generate Type II errors can be seen in figure 4-30 (a). The heuristic constraint for the low object defined by δ_{min} of 0.5 m is not valid for those objects, and part of those objects are less than δ_{min}. In this case, the UTF regards these objects as continuous terrain surfaces, rather than as discontinuous off-terrain objects. Figure 4-30 (b) shows another example of low object, which is not properly classified by the RTF filter. This

object is a gangway connecting a bridge to the ground, which appears in a green box of figure 4-29 (b). In figure 4-30 (b), lighter colour represents lower slope, whereas darker colour for higher slope. As can be seen in this figure, the boundaries of the low object cannot be completely isolated by slopes. Some parts of boundaries show similar slope properties to the ones of surrounding terrains, which means that the object is very smoothly elevated from the terrain surface in some parts. In this case, the RTF filter recognizes this object as the terrain surface since its height difference from the terrain surface is not obvious. However, the manual filtering classifies this as an off-terrain object.

Low objects are very difficult to remove even by manual processing. Some Type II errors are caused by the manual misclassification. Figure 4-30 (c) shows those errors which correspond to yellow circles in figure 4-29. In particular, Area-3 and Area-4 include large amount of Type II errors by the manual classification since their scene complexity is very high.



(a) low objects

(b) gangway

(c) manual classification errors

Figure 4-30. Type II error sources of RTF filter; red dots represent the Type II errors

4.5 Summary

In urban areas, there are some difficulties which often make the LIDAR filter fail to properly classify LIDAR points. These filtering difficulties are related to the object variety and complexity. This chapter evaluated the filtering performance of the RTF filter described in the previous chapter using two full scenes of LIDAR DSMs and analyzed filtering errors produced by the RTF filter in relation to object types.

In §4.2, the RTF filter was applied to a suburban area using the Shrewsbury data. Some data pre-processing methods such as thinning and tiling were proposed to remove the points which are too closely located, in order to properly construct a TIN as well as to reduce computational time. The quality of the terrain surface reconstructed by the RTF filter was visually investigated with support of orthorectified airborne imagery and the OS Landline.Plus® data. The result has shown that the RTF filter can clearly remove significant off-terrain features such as trees, building, and bridges, while it can successfully preserve geomorphologic shapes of embankments of railway and river. In comparison with the OS Landline.Plus® data, it has been confirmed that most of the on-terrain points classified by the filter are located outside the building vectors. This result suggests that it would be desirable to apply the RTF filter to a building detection application, which will be investigated in chapter 6.

Three KGPS profiles were used as the ground truth data for assessing the vertical accuracy of the DTM produced by the RTF filter. The KGPS profiles were projected on the filtered DTM, which was interpolated in a grid format by IDW (Inverse Distance Weighting) algorithm, and vertical differences between both datasets were measured. The result has shown that the vertical accuracy of the filtered DTM is 0.15 m in RMSE, which is almost compatible to the interpolation errors produced by IDW algorithm (~ 0.12 m). This suggests that the RTF filter can automatically reconstruct the DTM from a cloud of LIDAR points with high quality, and the generated errors by the filter remain only at random.

However, the quality assessments using the Shrewsbury data have also shown that the filter has its limitations in removing low objects with small extents. Some improvements of the RTF filter to overcome this problem were presented in §4.3.

Instead of simply lowering a height threshold, a different height threshold was adaptively selected to remove the low objects depending on a minimum size the of underlying local plane terrain model. This adaptive strategy makes the filter focused on the detection of terrain slope variations over larger areas, whereas focused on the detection of low objects over smaller areas. Some additional improvements of the RTF were also implemented to make the filter more adaptive to a high terrain complexity in urban environments. To this end, the RTF filter has been improved to classify generic terrain surfaces into more detail terrain classes including "disconnected terrain" and "bridge terrain", and to apply different filtering algorithms according to the terrain classification result. The modified RTF filter was applied to the Shrewsbury data. The results show that the filter can remove more low objects than the filter before the modification by 9.4 %, while it can still recognize bridges as off-terrain features.

In §4.4, the modified RTF filter was applied to a complex urban scene using the OEEPE LIDAR DSM. The filtered DTM has demonstrated that the RTF filter is useful to eliminate a variety of off-terrain features and preserve complex terrain landforms in urban environments. Four sub-scenes, which contain a high degree of object complexity and terrain discontinuity, were chosen to evaluate the quality of the filtered DTM in comparison with manually filtered reference data. The analysis showed that the RTF filter generated 0.94% for Type I error, whereas 6.75 % for Type II error. The Type I errors were produced in relation to terrain discontinuity. A very small amount of Type I errors suggests that the filter can effectively deal with a high discontinuity of terrain surfaces. The Type II errors were mainly generated by several factors; the misclassification of low objects and the difference of filtering rationales between the RTF filter and the manual classification. The mean values of the height difference between the filtering errors and the reference terrain were measured as 0.5 m for Type I errors and 0.88 m for Type II errors. This result suggests that the RTF filtering errors do not significantly degrade the overall quality of the terrain model reconstructed, and are only limited to very low objects less than 1 m in height from the terrain surface. The improvement to the RTF filter described in this chapter will be employed for generating terrain surface, and classifying non-terrain objects in more detail, and finally localizing building objects in chapter 6.

CHAPTER 5

BUILDING EXTRACTION FROM MONOCULAR IKONOS IMAGERY

5.1 Introduction

In general, building boundaries comprise various types of geometric features, i.e., point, line, and curve. Based upon that, most automatic building extraction systems start by obtaining those features from remotely sensed data by well-known low-level feature extracting algorithms. However, each extracted feature itself does not represent an object as a whole. Hence, the extracted image features must be transferred or mapped into a structured description approximating to a building shape. There may be two ways to convert those image features into an abstract representation of a building object, i.e., data-driven or model-driven method. This classical categorization of middle-level vision process is given, depending on how much the process relies on the use of semantic knowledge of the building object.

The data-driven approach performs the building extraction process without *a prior* knowledge of the building type or its specific structure, and emphasizes on the side of representing wide variations in building shapes. However, the data-driven method may fail where all significant features of a building object cannot be extracted in low-level vision algorithms, due to poor radiometric quality, shadow casting, or partially occluding parts by other objects. This problem can be solved by replacing the feature-based low-level processor with a region-based algorithm, which results in a complete description of object boundary. However, the region-based segmentation algorithm

frequently generates building boundaries with no regular geometric shape, which is not suitable for the purpose of delineating precise building boundaries.

In contrast, the model-driven method heavily uses pre-specified geometric constraints of building objects in a feature grouping process, such as parallel, orthogonal, and their combinatory properties. A significant feature, but omitted in low-level feature extraction can be efficiently recovered by the pre-assumed building constraints. Although the model-driven method allows more accurate and robust building extraction with a minimal number of features, the performance is sensitive to the quality of feature extraction, and the removal of distracting features is arbitrarily determined by pre-defined threshold, i.e., length or gradient magnitudes of extracted features. In addition, there is limitation representing generic building shape due to the restriction of geometric building constraints, and it is a difficult task to pre-design suitable feature grouping rules according to the data characteristics, such as radiometric quality, resolution and content complexity (rural, or urban).

This chapter exploits a hybrid feature-grouping method, in which advantages of both data-driven and model-driven approaches are efficiently combined, and provides a technique extracting building boundaries from monocular IKONOS imagery. The proposed feature grouping technique is based upon the assumption that building shape is composed of convex polygons, where each polygon represents a homogeneous photometric property. Those polygons are obtained when an initial polygon is recursively intersected by linear features extracted. The resulting polygons are piecewise homogeneous as they are encompassed by discontinuities, i.e., lines, and there is no linear features remained inside them. As polygons-bounded linear features are merged by a pre-defined homogeneous predicate, building outlines with regular geometric shape are finally generated.

The proposed technique is based upon a semi-automatic approach, where individual buildings are localized by the user as each building is bounded by a rectangle. A polygonal segmentation by extracted features is implemented in a BSP (Binary Space Partitioning) tree. An optimality of the BSP function is achieved as distracting linear features are removed by a global analysis of dominant building orientations, and remained lines are involved in the partitioning function according to a "significant"

order; a line that maximally responds to edges is used in the partitioning function prior to the rest of the lines.

The next section discusses problems posed in cue extraction and grouping process when monocular IKONOS imagery is solely used. In following sections, the aforementioned building extraction technique is presented in detail. Finally, the limitation of proposed technique and conclusions are discussed with experimental results.

5.2 The statements of problems

Most of building object is a regional feature. There may be two ways to obtain the building region; constructing region from extracted features (feature-based approach) or detecting features from existing regions (area-based approach). In this section, perceptual organization and region growing methods are considered as representative techniques of both approaches noted above, and various pros and cons of two techniques are discussed.

5.2.1 Perceptual organization

Perceptual organization has been widely adopted as a middle level of vision process for the building extraction problem. In the last few years, some of the results based upon this technique have successfully demonstrated that perceptual organization can be recognized as an essential element in future building extraction systems. Basically, this organization scheme starts by seeking the local salient features in the image, and then groups these cues to make the building hypotheses based on a set of reasonable causes, like Gestalt criteria with minimal domain knowledge (Boyer & Sarkar, 2000).

However, the most difficult problem, regarded as a bottleneck in this bottom-up process, is always caused by the fact that the low-level feature extraction algorithms do not provide sufficient cues for building reconstruction, especially in the complex scenes such as aerial or satellite imagery. Due to this difficulty, even in the parametric building model of the box-type or its combination (Kim & Muller, 1999; Lin & Nevatia 1998; Shufelt 1996), which requires only two or three-paired lines, an "accidental" cause is often considered to create the building hypotheses in a way of extending the line within certain pre-defined query space.

In order to circumvent this problem, a number of research efforts have been made to look for a complementary method to enhance these insufficient cues, which were partially or completely lost when the real object space was projected onto the image space. A single cue (e.g., line or corner) in 2D was enhanced, either in quantity (i.e., density) or in quality (i.e., accuracy and fragmentary level) using multiple images taken from different image frames (Ameri, 2000; Baillard et al., 1999; Fischer et al., 1998). On the other hand, cue integration was performed, in which extracted cues have multiple properties, such as colour constancy, brightness homogeneity and texture regularity (Fischer et al., 1998; Henricson, 1998). The other issue is to increase cue dimensionality, in which 3D corners (Fischer et al., 1998), 3D lines (Baillard et al., 1999; Haala & Hahn, 1995) or 3D planar polygons (Ameri & Fritsch, 1999) were employed for the building reconstruction purpose. However, these previous efforts to enhance the cue properties for detecting more complex building shape mainly focus on a particular range of image resolution, i.e., sub-metre resolution, and have been less concerned about effects, which may be caused when the scene content and its spatial resolution is changed (Mayer, 1999). Consequently, using lower spatial frequency such as 1-metre resolution degrades the object resolving power and brings more complex problems for reconstructing the buildings.

There has been little work carried out in recent years using 1-metre image resolution to develop methods of building detection in high-density urban areas. Muller et al. (1997) reported a problem caused by insufficient line density when a line relation graph searching method (Kim & Muller, 1995) was applied to simulated data compatible with 1-metre resolution imagery. To work out this resolution effect, several systems have been developed for making the building detection process use the perceptual cues as little as possible. A semi-automatic system for building reconstruction in urban areas was suggested, in which a generic shape of building roof was detected in 1-metre airborne imagery using Active Contour Model, known as "Balloon" (Yoon et al., 1999). The system requires an operator to select a seed point, where the "Balloon" model starts. From the selected seed point, a contour is expanded by an external guidance force, and terminated as the contour is maximally corresponded around building boundaries. Finally, the building hypothesis is generated by aggregating the rectilinear lines located nearby a building region, which is constructed by the balloon model. Considering the low signal-to-noise ratio and spatial frequency of IKONOS, Park et al. (2000)

delineated a rectangular shape roof in a high-density residential area as directly creating a conjugate anti-parallel line over the roof using a line-rolling algorithm.

In the above mentioned research works, the following difficulties are commonly raised when generating building hypotheses using 1-metre image scale.

- Its low signal-to-noise ratio disturbs the extraction of geometric primitives at the bottom-level of vision process. For this reason, minimum cue density required for conventional line-to-line or line-to-junction analysis used in building hypotheses generation and their verification process cannot always be guaranteed and also geometric parameterization of these primitives is perturbed by the noise.

- The low spatial frequency of IKONOS imagery lowers the object resolving power. Absolute object scale is represented with relatively smaller pixels compared to sub-metre image scale, and object density in image space becomes denser. This higher object density means shorter physical distances, for example, between building and neighbouring buildings or other cultural features, and makes it more difficult to separate a single object from surrounding ones. It degrades the reliability of the "proximity" rule, mainly adopted in perceptual organization scheme, and constructs extraneous building hypotheses, which may result in substantial detection errors. In consequence, more verifying cues are needed to minimize these errors. However, these cues are not going to occur in 1-metre resolution data. Also effective domain knowledge, assuming that the brightness of the building roof-top is always brighter than that of the ground, or a sharp corner of the shadow cast by a building structure is consistently detectable on the ground, cannot be used reliably as highly weighted evidence for building detection.

5.2.2 Region-growing

The perceptual organization described in the previous section is performed as extracting features defined by significant local change in the image, and the following grouping process construct a building region. However, those features can be defined in a different way as being located between homogeneous regions. Based upon that, region

growing techniques segment the whole image into sub-regions of maximum homogeneity by aggregating similar pixels and then, merge regions satisfying a pre-defined homogeneity predicate. As a result, boundary features made between segmented regions can be obtained. The homogeneity criterion is an important property in region growing, which can be based on gray-level, colour, and texture. A number of excellent researches in this technique can be found in Adams & Bischof (1994), Beveridege et al. (1989), Gambotto (1993), Pavlidis & Liow (1990), and Zhu & Yuille (1996).

Compared to the perceptual organization method, the region growing technique is generally better in noisy image, where features are extremely difficult to detect. In addition, it provides closed object contours so that regional attributes, such as area, perimeter, and compactness, and radiometric property (average gray level and variance), can be driven. On the other hand, the topology of regions, boundaries, and nodes is implicitly acquired during the process. These benefits provide useful knowledge to either localize buildings (Lee et al., 2003) or understand complex building roof structures (Ameri & Fritsch, 2000; Haala & Andres, 1997). However, the region growing technique can lead to the generation of "blobs", whose boundaries are ragged with no regular geometric shape and do not always coincide with edges. This problem is caused by the fact that no explicit model for the boundaries is employed in the process, which is not suitable for the purpose of delineating building shape. To circumvent this drawback, algorithms for integrating region growing and feature detection were suggested by Gambotto (1993) and Pavlidis & Liow (1990). However, the performance of those methods entirely depends on the quality of feature extraction, and it may be problematic when IKONOS imagery is used. An alternative post-processing without using features was devised, where resulting ragged boundaries are regularized as favorable geometric constraints are coded in MDL (Minimum Description Length) optimization method (Weidner & Forstner, 1995; Wiman, 1995) or noisy polygons are smoothed by Douglas-Peucker method as in Steinle & Vogtle's work (2001). However, these "algorithmic" modifications of noisy building boundary still do not coincide with "real" object edges.

5.3 System outline

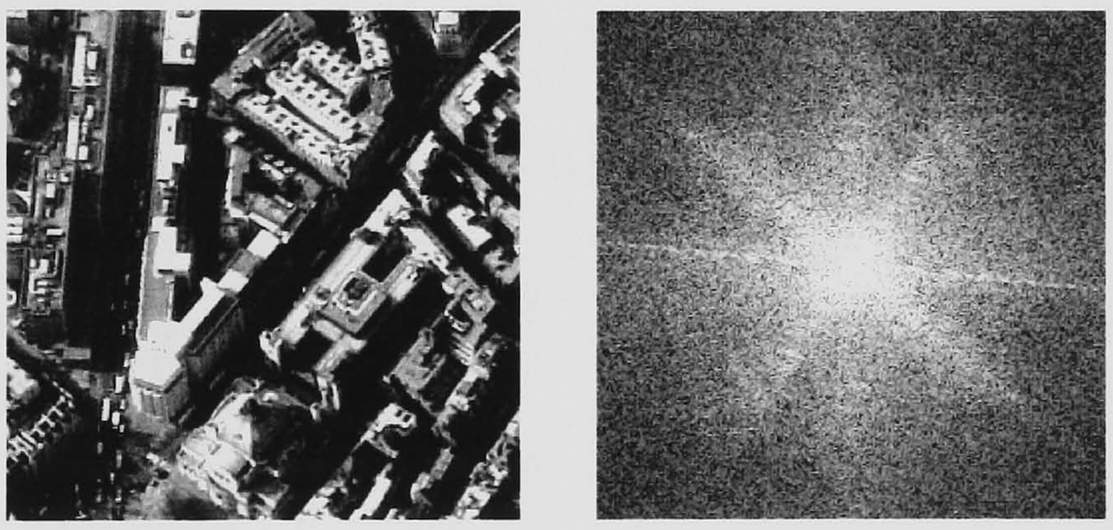
The proposed system aims to delineate building boundaries from monocular imagery. In the light of this research purpose, linear features extracted from the image must play a

very important role to place exact building location. The problem is to determine i) which features belong to building objects, especially to building outlines, and ii) how topological relationships between lines are made to construct a building polygon. In 2D image space, the former question can be answered by removing distracting lines, whose length is less than certain threshold, or gradient magnitudes look like noisy features. However, the criterion used is arbitrary and "significant" lines of building outlines may be seriously lost in the particular case that the fragmentary level of line extraction is high such as in IKONOS image. For the latter problem, the perceptual organization method described in the previous section can be considered. However, on the one hand, the low signal-to-noise ratio prevents a line detector from providing sufficient cue density; on the other hand, the limited resolving power of 1-metre resolution lowers reliability of the perceptual organizing rules such as proximity and collinearity. To overcome the aforementioned difficulties, the proposed building extraction system considers the following aspects:

- At the bottom level, the cue generation process allows us to extract noise-like lines by the use of rather "loose" criteria for filtering distracting lines. Instead, the "significant" lines, belonging to building outlines, are selected through a global analysis of dominant building orientation.
- A cue inference method, recovering fragmented lines or reconstructing missed lines, is implemented by a Binary Space Partitioning (BSP) method (Fuchs et al. 1980), where extracted lines are "optimally" extended by recursive line intersection.
- As a result of BSP function, a topology made between extracted lines is implicitly constructed, and whole image is divided into a set of convex polygons with rectilinear geometry.
- It is assumed that a generic shape of building is composed of convex polygons with similar gray-level. Starting from a seed polygon located around the centre of a building roof, polygons with similar homogeneity are merged and final building outlines are reconstructed.

As can be seen in figure 5-1(a), interpreting individual lines in the image is seriously disturbed due to the rich context of high urban density area. However, dominant directions in a building cluster can be easily observed in a compressed FFT (Fast

Fourier Transformation) domain (see figure 5-1(b)). This is caused by following two factors. It is not the usual case that most buildings are aligned to one or two fixed city grids as employed in Jaynes et al.'s work (1995). However, building objects show a "regular" directionality in their orientation, rather than random, when scene complexity is resolved in a focused region with certain size. On the other hand, in a building cluster, each building object is aligned piecewise with the other neighbouring building sides in approximately the same directionality. This coherent spatial arrangement augments a "group saliency" of dominant building orientation. Thus, a global analysis of dominant building orientation can be used as a useful knowledge, which helps the process remove distracting lines and recover fragmented or missing lines.



(a) (b)

Figure 5-1. FFT power spectrum of IKONOS sub-scene in London; it shows the difficulties to separate a single object from surrounding features due to its scene complexity (a); however, behind its complexity, we can observe coherently aligned building clusters, whose characteristics can be represented by salient directionality in the compressed FFT power spectrum (b).

Firstly, an individual building object is localized by the user and a targeted building is bounded by a rectangle, which allows it to include a certain number of neighbouring features such as roads or buildings. The next step is to globally analyze a set of building orientation angle. These dominant angles are calculated by an analytic function of Fourier transform introduced by Yang et al. (1996). To this end, straight-line primitives are extracted using the Burn's algorithm (Burns et al. 1986). Global information of building orientations is given through a local Fourier analysis of every edge pixel that consists of the detected line segments. By that, geometric parameters of building boundaries are predetermined. A feedback from this high-level information to low-level

vision processing makes the Burn's algorithm focused on line parameters related to the predetermined building orientations only. As a result, a number of extraneous line segments to generate problematic building hypotheses are reduced, and line parameters perturbed by the noise are regularized (see figure 5-2 (a) and (b)).

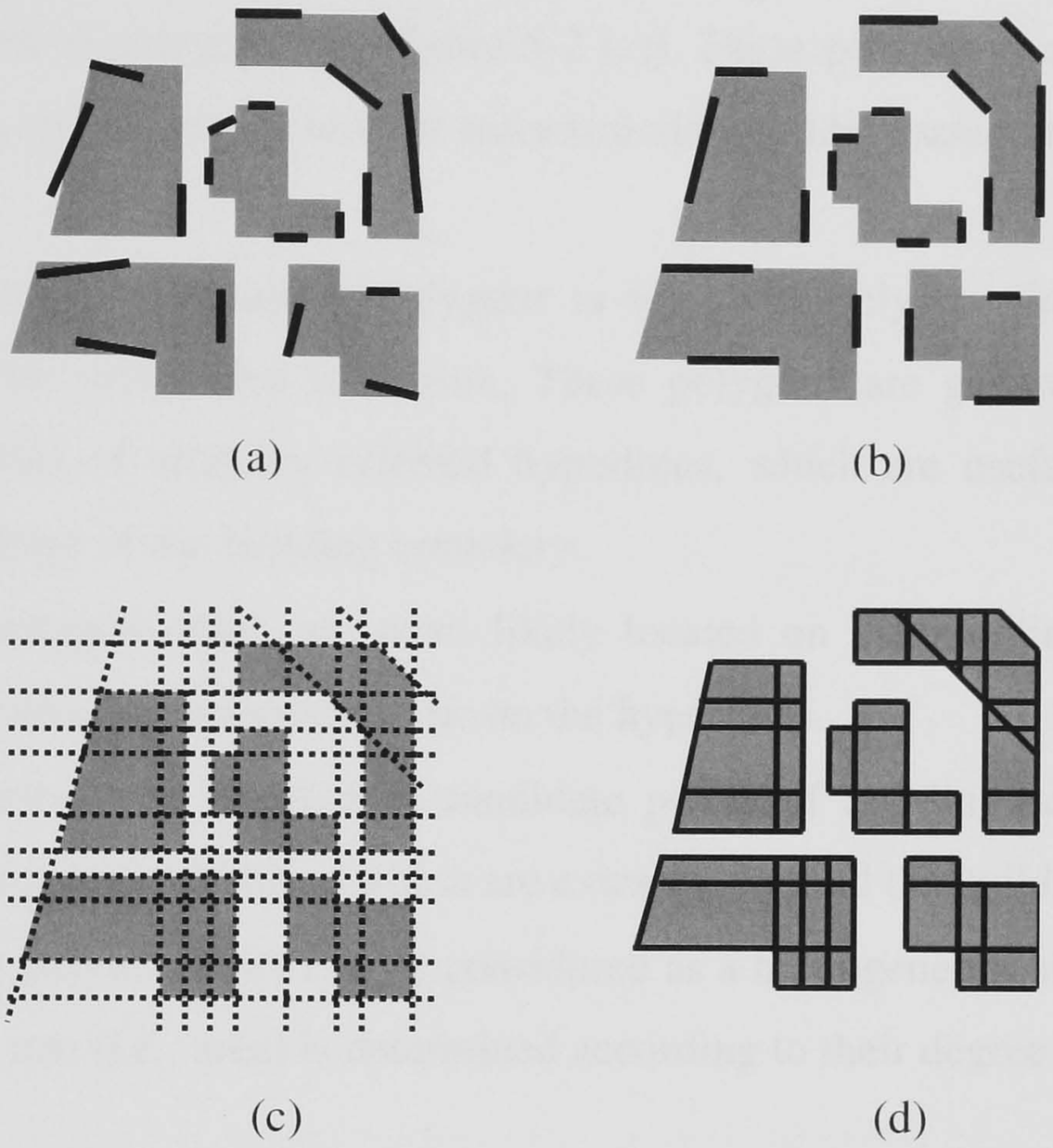


Figure 5-2. Schematic diagram of the partitioning of a image domain into BUSes; (a) noisy line segments extracted by conventional Burns algorithm; (b) regularized line segments using the analysis of global dominant orientation angle information; (c) a polygon space generated by the recursive partitioning of the image domain using regularized line segments, and (d) reconstructed building objects found by grouping only "interest" polygons related to the building objects.

However, due to weak local saliencies of the building boundary in 1-metre resolution imagery, line cues constructing building outlines may be completely lost or highly fragmented. In the suggested system, completely missed lines are reconstructed by inferring them from detected line cues of neighbouring building boundaries which are aligned in approximately the same directions, and also partially extracted line are recovered by extending themselves along predetermined building orientation. To achieve this cue inference process, a Binary Space Partitioning (BSP) method is employed, which recursively partitions entire image space by line segments. A rectangle

drawn by the user is used as an initial convex polygon, and extracted line segments are employed as hyperlines to halve that initial polygon. Those hyperlines are sequentially selected in a "significant" order. As the most "significant" line is selected and intersected with the initial polygon, two sub-convex polygons are generated. This binary partitioning process continues until no child polygon is generated by linear features, and ends up with a set of polygons (see figure 5-2 (c)). These polygons have the following characteristics as an elementary unit for reconstructing building structures:

- The geometry of resulting polygons is a convex polygon since initial image space to be subdivided is convex. These polygons are generated through the intersections of arbitrary oriented hyperlines, which are useful to represent a generic shape of the building boundary.
- All the polygon edges are most likely located on the building boundary; all discontinuities are restricted to lie on the hyperlines.
- Their vertices can be used as candidate points of building corners; these are intersecting points of lines, which are extracted around the building boundaries.
- Resulting polygon faces can be considered as a homogeneous surfaces and their sampling rate (i.e., area) is determined according to their degree of homogeneity.

For convenience, each polygon generated by BSP is now called as a BUS (Building Unit Shape). A BUS space is generated to represent a BUS adjacent graph, where each node represents a BUS and each arc means the connectivity between neighbouring BUSes. Since the resulting BUSes are piecewise homogeneous encompassed by discontinuities, a BUS grouping process eliminates spurious lines besides building boundary by merging statistically similar BUSes (see figure 5-2 (d)). As a result, building outlines are reconstructed.

The BUS grouping has several advantageous over conventional region growing algorithms; i) it does not generate noisy boundaries since the BUS is generated directly by edge components of object boundaries, not by pixel-based similarity test; ii) it is statistically more reliable to measure a gray-level similarity as the BUS consists of a number of pixels, not a single pixel; iii) the resulting outlines follows globally driven building structures. It is difficult to achieve these properties in a conventional region

growing framework. Figure 5-3 illustrates the aforementioned method for extracting building outlines from monocular IKONOS imagery.

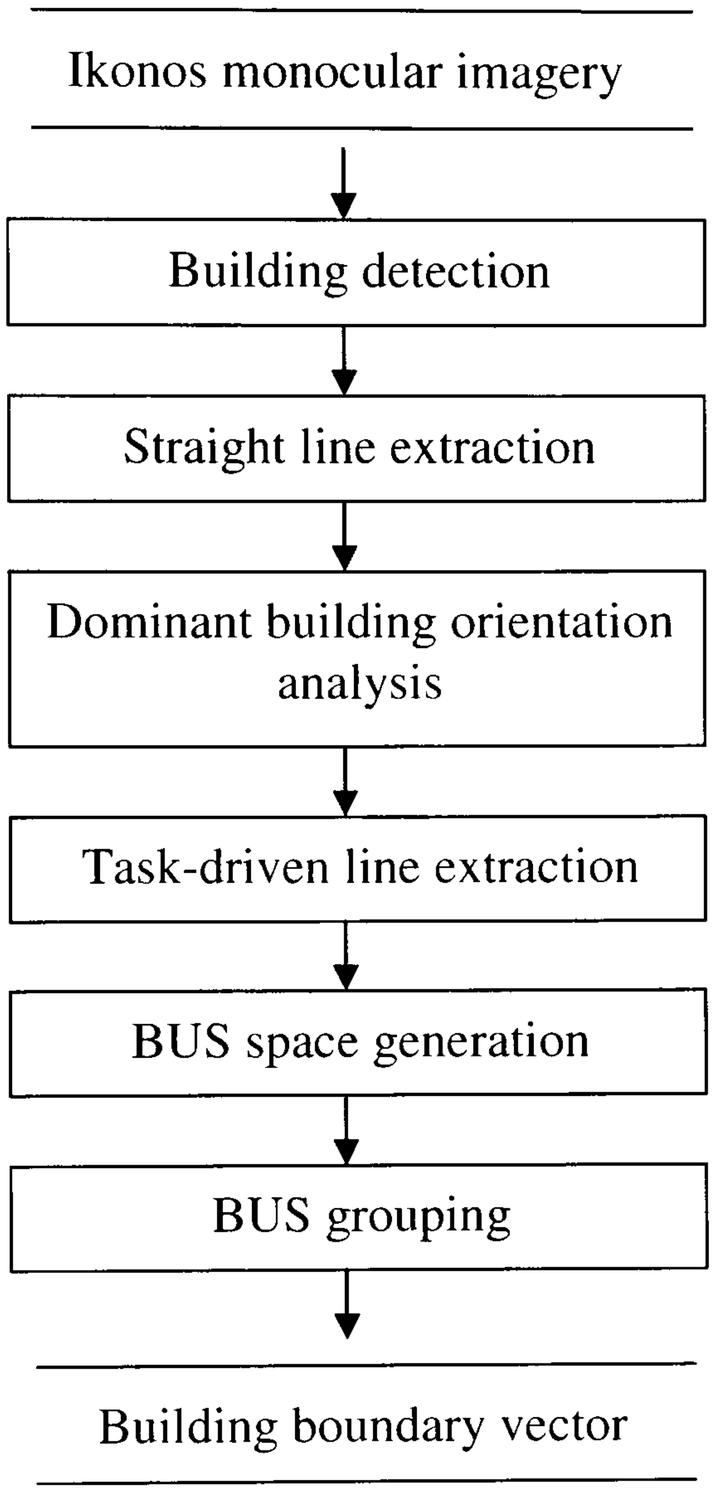


Figure 5-3. Schematic diagram of proposed building extraction system

5.4 Building Unit Shape (BUS) organization

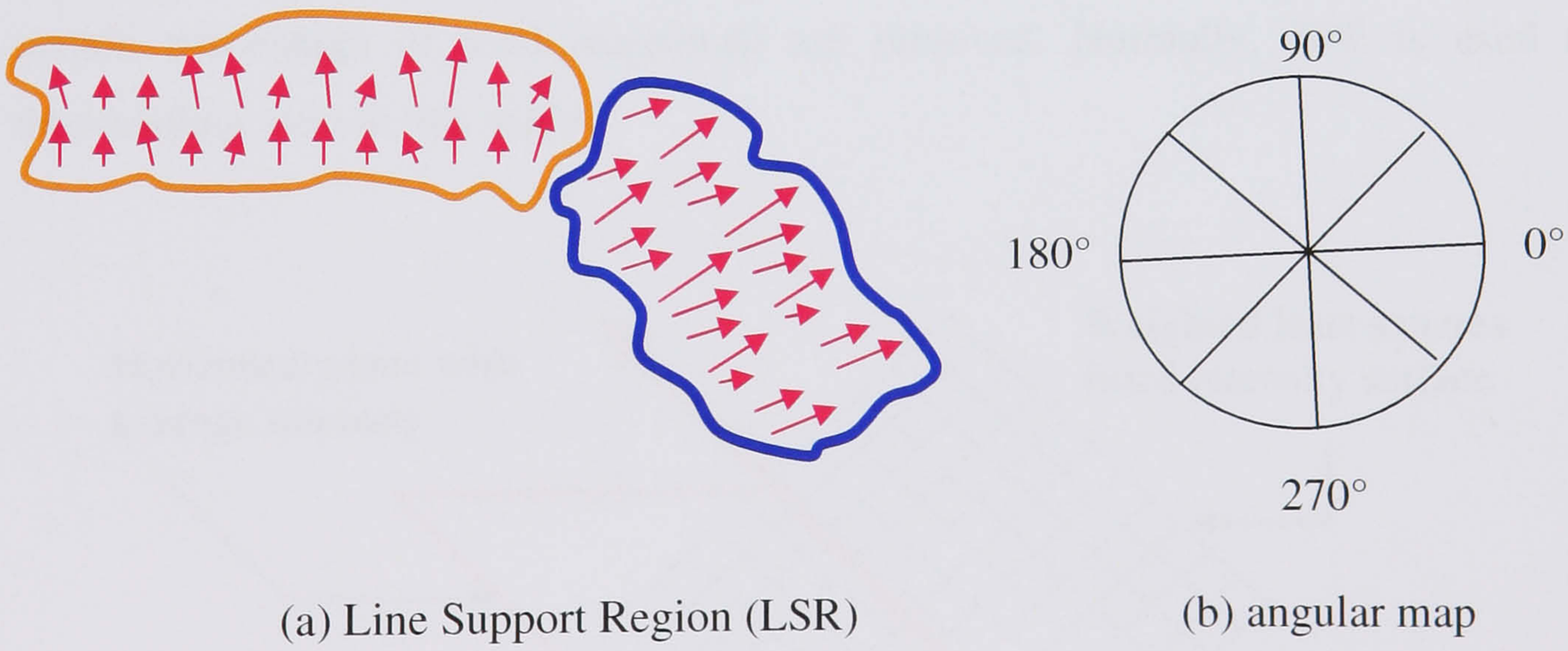
This section describes each algorithm shown in figure 5-3, mainly presenting how linear features extracted by a low-level of machine vision algorithm are converted into polygonal segments of the image given, and finally how building outlines are obtained by them. The suggested algorithm is tested over single IKONOS imagery.

5.4.1 Straight line extraction

It is assumed that the saliency of each building boundary substantially contributes to determine a dominant pattern of orientation angle in the scene and most of the building boundaries consist of rectilinear lines. Thus, the extraction of straight linear features is

required. For this purpose, a straight line extractor suggested by Burns et al (1986) was implemented. Many edge and line extracting algorithms rely on a two-step process; i) detection of local strong edges indicating significant features in the scene, and ii) aggregation of edge pixels into more globally detected lines on the basis of various grouping criteria. The problem encountered with the aforementioned approach is caused by the fact that decision of extracting and grouping edge pixels is locally processed. Burns et al. (1986) proposed a global line extracting algorithm, which aims to find straight lines in complex images, particularly those lines that are long but not necessarily of high contrast. Hence, the Burns algorithm is suitable for the current research goal to detect straight lines from IKONOS imagery.

The first step of Burns algorithm is to calculate local gradient orientation for each pixel and then to group pixels having the same gradient orientation into a region, namely "Line Support Region" (LSR) (see figure 5-4(a)). To this end, the 360° range of gradient directions computed is arbitrarily quantized into 8 sectors with 45° regular interval, and each gradient vector is labelled according to the partition into which it falls (see figure 5-4(b)).



(a) Line Support Region (LSR) (b) angular map

Figure 5-4. Line Support Regions (LSR) and angular map in the Burns algorithm

A simple connected-components algorithm is used to form unique region labels for groups of adjacent pixels with the same orientation label. For this purpose, a connected-component algorithm suggested by Lumina et al. (1983) was implemented. In order to reduce quantizing errors due to the fixed partitioning scheme, the first partitioning table

is shifted by 22.5°, by which the second LSR map is calculated. Now each pixel has the two representations of its belonging LSR. The LSR considered best for the pixel is the one that provides an interpretation of longer line. Each pixel votes for one of two belonging LSRs which has larger size and finally, the LSRs only remain if they have the "support" larger than 50% from their member pixels. Burns et al. (1986) suggested a 2x2 mask to calculate gradient orientations. However, in current implementation, a 5x5 Sobel-like mask was used to maximize the low signal-to-noise ratio of IKONOS imagery followed by Yoon et al.'s work (1999).

The next step is to approximate the intensity surface associated with each LSR by a low-order planar surface fitting. The parameters of this plane are obtained by a weighted least squares fit to the intensity surface so that the pixels with stronger gradient values contribute more to determine the planar surface parameters. Then, a straight line is obtained as the intersection of the obtained surface plane with a constant horizontal plane represented with average intensity of LSR as shown in figure 5-5. In current implementation, nose-like pixels are filtered in advance not to be involved in the computation of two planes. Followed by Canny (1983), a gradient magnitude histogram is calculated from each LSR. When the gradient magnitudes are sequentially accumulated from the lowest value, the pixels having the gradient value less than certain percentage of total magnitude are removed. Normally, 30% is used as a thresholding ratio in this study.

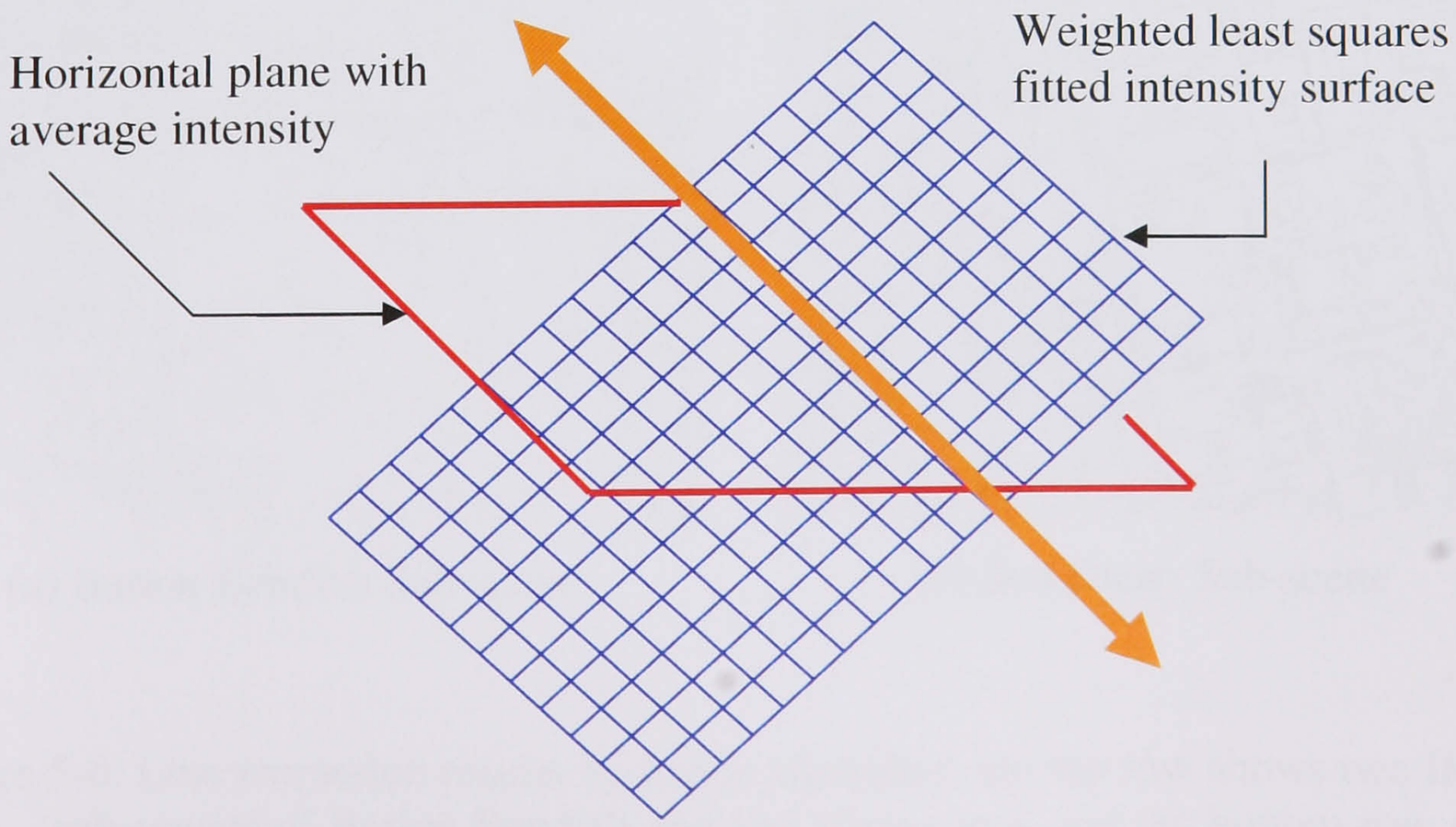


Figure 5-5. Straight line extraction in the Burns algorithm

The implemented Burns algorithm was tested to extract straight lines from a range of test images as a part of ALFIE (Automatic Linear Feature Identification and extraction) project (Jackson, 2003). Sub-scenes were extracted from the Barton Bendish and San Diego IKONOS images, 600 x 600 pixels in size. The scenes were selected in order to cover a variety of land uses (Rural, Urban, residential house and industrial building). The parameters of Burns algorithm such as image noise level and extracted line length were predefined as 0.3 and 10 metre respectively. The results of the algorithm are in figure 5-6.

(a) Barton Bendish Sub-scene

(b) San Diego Sub-scene

Figure 5-6. Line extraction results by Burns algorithm; the top row shows two IKONOS sub-scenes of Barton Bendish and San Diego area, and the bottom row presents line extraction results by Burns algorithm.

From the results presented in figure 5-6, it can be concluded that the Burns Algorithm is appropriate for extracting building across IKONOS images, and resulted in good line extractions for building footprints and edges. In the Barton Bendish sub-scene (figure 5-6 (a)), straight roads in images are extracted very clearly, apart from where obstruction is caused by the presence of trees. Individual small buildings (houses) have not been delineated clearly, but garden boundaries have been extracted well. Larger buildings in right hand side of image have been delineated. In the San Diego sub-scene (figure 5-6 (b)), buildings are delineated extremely well, although generally more than one line has been extracted for each building edge (possibly because building footprints and roof tops have been detected). Most buildings are flat-roofed, but where peaked roofs are present, some detail of roof shape has been picked up. Although the algorithm successfully extracted relatively large buildings such as those found in the San Diego scenes, it did not work well for extracting smaller buildings (e.g. houses) that had edges as small in length as 10m. The Burns algorithm is therefore less suited to extracting small buildings (size ~10 m). Other drawback noticed in both results is that the line geometry of extracted lines does not exactly coincide with "real" building outlines in some cases, that the contrast between building and ground is not very high or noisy artifacts, are densely scattered over certain area.

5.4.2 Dominant orientation angles and task-driven line extraction

A dominant orientation angle in the scene can be seen when significant features are aligned with regular directionality. This information has been used in some building extraction system for either pruning distracting features or grouping linear features with pre-designed structuring rule. Kim et al. (2001) used a length-weighted histogram of line angles from extracted linear features and selected the two highest peaks in the histogram as the dominant angles. Similar to this, Zhang et al. (2001) introduced a "Weighted Histogram of Angles" (WHoA), which employed a distance between edge pixels and straight lines as well as the line length for weighting the angle histogram.

In the current research, considering rather poor resolving power of IKONOS image and inherited shortcomings of the Burns algorithm, a more complicated method was exploited to obtain the dominant orientations. Although the Burns algorithm is well known to be less sensitive to the noise, the surface planar fitting for each LSR is basically a parameter estimation process relying on a weighted least squares and thus

the drawback of the least square method is inherited to the Burns algorithm, That is, in case the contaminated noise is larger than a certain amount or its statistical characteristic does not follow Gaussian distribution, estimated parameters are easily disturbed with that noise found in IKONOS imagery. For this reason, the lines extracted by the Burns algorithm only serve as candidate pixels to determine dominant orientation angles in the scene. However, those line parameters perturbed by the noise are regularized by the support of extracted dominant angles in later stage.

In this study, dominant angles from complex imagery are determined through a local Fourier analysis of edge pixels. For a strongly oriented intensity pattern along one direction, the power spectrum clusters along a line through the origin in the Fourier domain. By determining this line, a dominant orientation angle, θ, can be extracted. Yang et al. (1996) proposed an analytic function of Fourier transform of the image to measure anisotropism at each pixel for noise filtering of images. The major advantage of the function is to define oriented angle location of edge pixels in Fourier domain without using the actual computation of the Fourier transformation. Given a 2D Image, I, the analytic Fourier transformation function calculates an edge orientation angle, θ, as follows:

$$\theta(\mathbf{x}) = \frac{1}{2}\tan^{-1}\frac{\iint_{\Omega} 2I_x I_y dxdy}{\iint_{\Omega}(I_x^2 - I_y^2)dxdy} + \frac{2}{\pi} \tag{5-1}$$

where $\mathbf{x}$=(x,y) and Ω is a small size of neighborhood integrating first-derivatives, i.e. I_x and I_y. These single derivatives are defined as follows:

$$I_x = \frac{\partial I}{\partial x};\ I_y = \frac{\partial I}{\partial y} \tag{5-2}$$

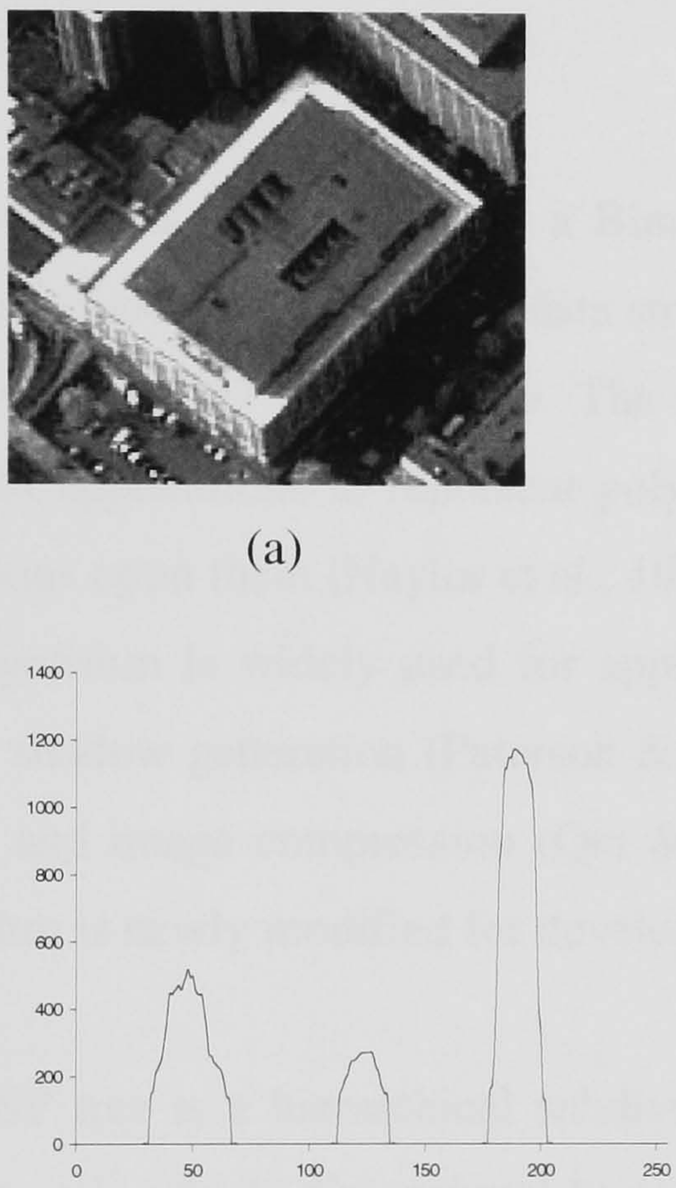
By Eq. 5-1 and 5-2, a set of dominant orientations, $\{\theta\}$, are calculated over line segments extracted from the Burns algorithm. In practice, a small radius of neighborhood Ω =5 and a 5x5 Sobel filter is used for calculating I_x and I_y. The remaining problem is to determine M dominant orientation angles, $\{\theta_d^i\}_{i=1}^M$, from $\{\theta\}$. Following a histogram-clustering scheme of Chabat et al. (1999), derived orientation angles, $\{\theta\}$, are subdivided into a histogram quantized by 256 discrete angular values and corresponding gradient magnitude of each orientation angle is accumulated in the histogram to make high gradient pixels contribute more to determine dominant angle peaks. Local maxima associated with dominant angle peaks, θ_d, are obtained after applying a simple average filter with the smoothing mask size of 16 angular units and thresholding noise-like angles less than 1% of the maximum gradient magnitude accumulated in the angle histogram. Finally, separated maximum peaks are found as dominant orientation angles from the gradient weighted angle histogram.

Once the dominant orientation angles, $\{\theta_d^i\}_{i=1}^M$, are obtained, line parameters disturbed by the scene noise are regularized by them. Suppose a line l is extracted by Burns algorithm and its line geometry is represented with θ_j and d_j, where d_j is the distance of the origin from the line, θ_j, express the angle between the edge and x axis. If θ_j is similar to one of dominant orientation angle, θ_i^d, within a tolerance of θ_{th}, $l_j(\theta_j, d_j)$ is replaced with $l_j(\theta_i^d, d_j)$. This is described as follows:

$$l_j(\theta_j, d_j) = x_j \cos\theta_d^i - y_j \sin\theta_d^i - d_j, if \left|\theta_j - \theta_d^i\right| \le \theta_{th} \tag{5-3}$$

In practice, θ_{th} of 10 angular units was used for constructing the above mentioned angle histogram. Figure 5-7 illustrates how the dominant orientation angles are obtained and thus, lines parameters are regularized. As can be seen in figure 5-7(b), straight lines are extracted by the Burns algorithm from an IKONOS sub-scene of figure 5-7(a), but those line parameters are disturbed by scene noises. After obtaining three dominant

orientation angles by the above mentioned local Fourier analysis and histogram clustering method, this angle information supports the line parameter regularization. As can be seen in figure 5-7(d), the process successfully works, where those perturbed lines are properly re-aligned to real linear features of the sub-scene.



(a) (b) (c) (d)

Figure 5-7. Illustration of extracting task-specific line segments using the analysis of global dominant orientation; (a) a sample image of IKONOS in London; (b) noisy line segments extracted by conventional Burns algorithm; (c) global dominant orientation angle histogram after adopting histogram-clustering method, and (d) re-extracted and regularized line segments using global dominant orientation angle information.

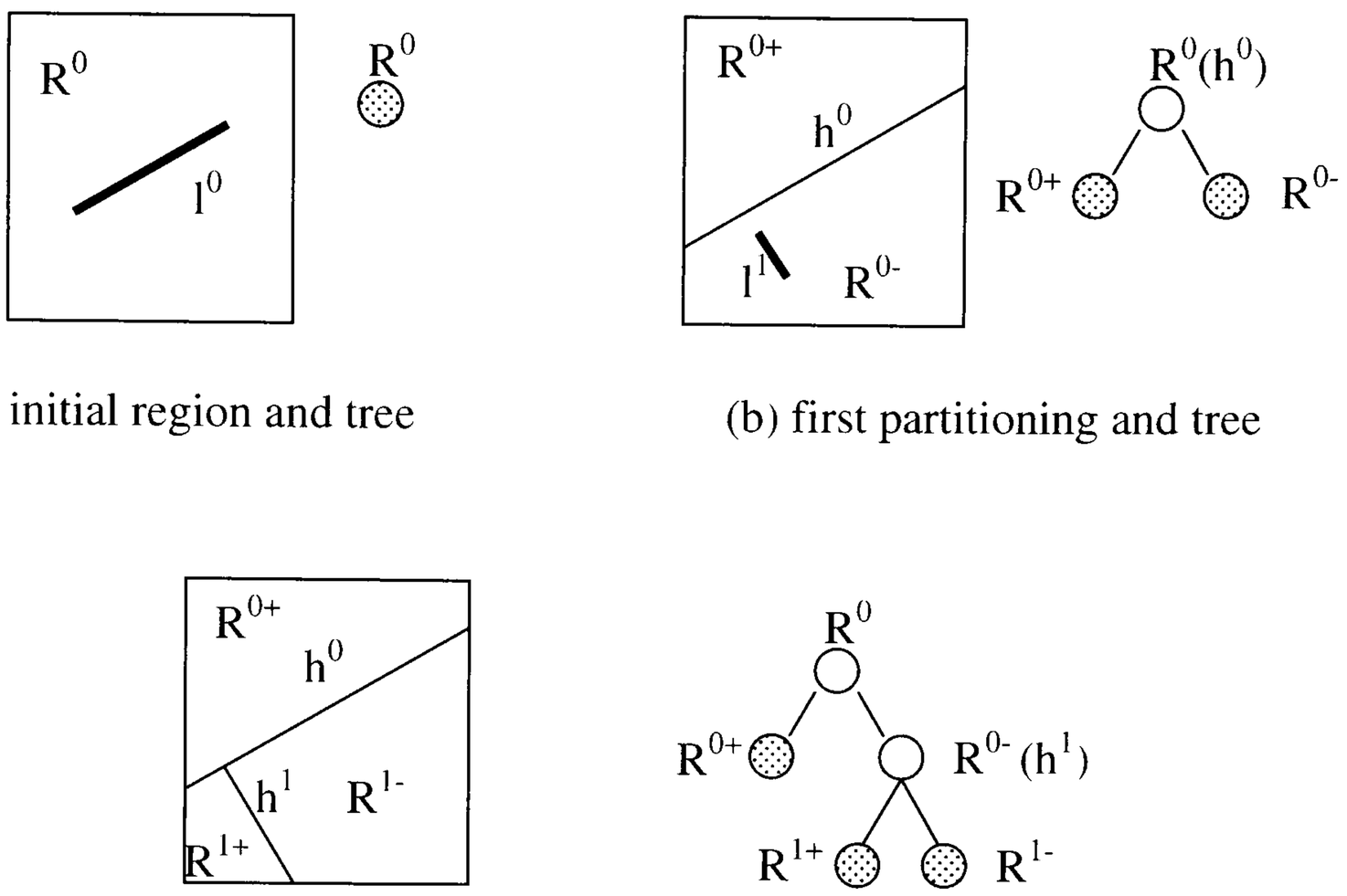
5.4.3 BUS space generation

This section describes a framework for middle-level of feature grouping, namely BUS (Building Unit Shape) space, where highly fragmented or missed building boundaries are recovered and topological relationships made between linear features are implicitly constructed during the process. The BUS space is generated by recursive partitioning of a region with extracted lines, which ends up with a set of convex polygons. Since a BUS with maximal homogeneity is generated by extracted linear features, this

polygonal partitioning method can be considered as a hybrid approach, where feature-based and area-based segmentation method are combined. Moreover, a cue inference recovering fragmented or missed line segments is achieved during the partitioning process, and results in a complete connection of neighbouring lines. After the partitioning process, the BUS space is represented as a polygon adjacency graph, where each node represents a convex polygon, i.e. BUS, and each arc means the connectivity between neighbouring polygons.

For constructing a BUS space, a Binary Space Partitioning (BSP) tree (Fuchs et al., 1980) is adopted as an efficient data structure to aid recursive partitioning of a region by a hyperline in 2D image space. The BSP tree was initially developed for computer graphics applications to represent polyhedra in D-dimensional space, and perform set operations upon them (Naylor et al., 1990; Thibault & Naylor, 1987). Currently the BSP tree algorithm is widely used for application ranging from motion planning (Tokuta, 1994), shadow generation (Paterson & Yao, 1989), image representation (Radha et al., 1991), and image compression (Qiu & Sudirman, 2001). In this study, this BSP tree algorithm is newly modified for developing a building extraction system.

The BSP tree is a hierarchical subdivision of n-dimensional space into homogeneous regions, using (n-1)-dimensional hyperplanes. For 2D case, the subdivision is created using arbitrarily oriented lines, known as hyperlines. Fig 5-8 illustrates this. Suppose that an initial polygon bounding a building object is R^0, and l^0 is an extracted line within R^0 (see figure 5-8(a)). When l^0 is selected as a hyperline h^0 to subdivide R^0, h^0 is now defined by an expression of the form $h^0 = a + bx + cy$, where a, b, and c are line parameters defining h^0. The set of points in space that make $h^0 > 0$ define the front (positive) half-space of R^0, i.e. R^{0+}, while those that make $h^0 < 0$ define the back (or negative) half-space of R^0, i.e. R^{0-}. The points $h^0 = 0$ are on the hyperline h^0 (see figure 5-8(b)).



(a) initial region and tree

(b) first partitioning and tree

(c) recursive spatial partitioning and binary tree

Figure 5-8. BSP tree construction.

The BSP tree is stored in a binary tree structure. As seen in figure 5-8(a), an initial building polygon R^0 corresponds to a root node in the BSP tree. This tree is subdivided into R^{0+} and R^{0-} by subdividing R^0 with h^0. As an internal node, the root node holds a hyperline h^0 that partitions the region at that node into positive and negative sub-regions. The two sub-regions are represented by "children" of the node (see figure 5-8(b)). A leaf node, in contrast to an internal node, holds no hyperline and corresponds to an unpartitioned region. This process is repeated again recursively until no leaf node is generated by BSP function.

However, there are usually many line segments in a region, and the construction of the BSP tree is sensitive to the selection of hyperline used for partitioning a given region. Thus, a selection priority of hyperlines must be determined. To this end, a magnitude of edge correspondence is used as the criterion for sorting the hyperlines in a "significant" order, where a hyperline with maximal edge correspondence is determined as the most significant one, and finally partitions the region.

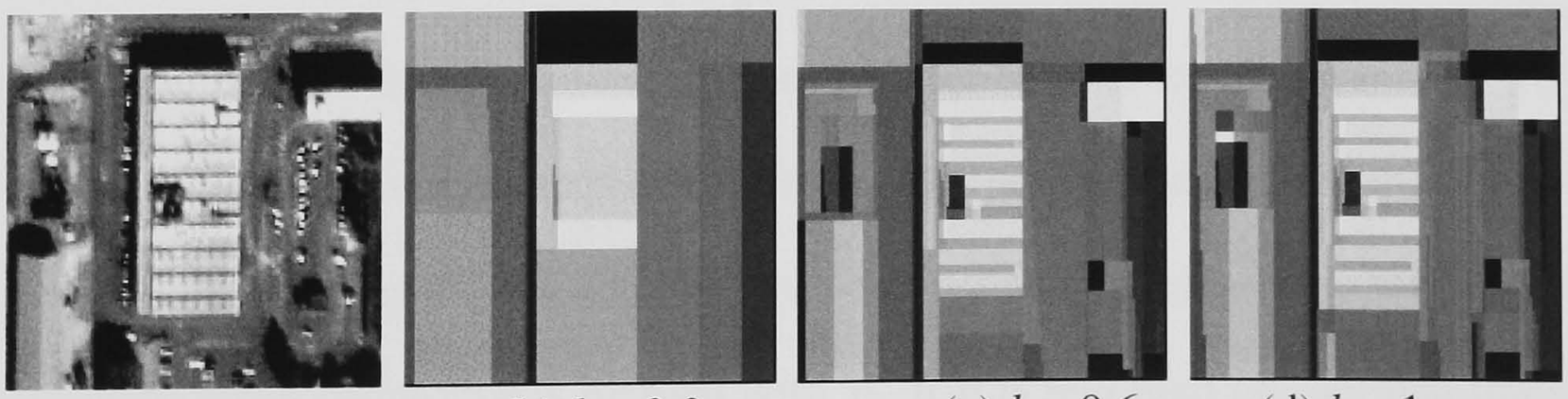
Suppose that N line segments $\{l^i : i=1, \ldots, \mathrm{N}\}$ are extracted in R^0, which are considered as N hyperline candidates $\{h^i : i=1, \ldots, \mathrm{N}\}$ to divide R^0. The gradient magnitude is measured in average over all of the pixels located where a hyperline h^i intersects with R^0, which is stored as a cost value of edge correspondence into a sequential list of hyperlines. This line list is sorted in a "significant" order so that a hyperline h^i passing through the largest number of edge pixels over the entire image domain R^0 is located on the top of the sequential line list.

After setting up the hyperline list, a hyperline h^0 with the highest cost value is selected to partition the whole image domain R^0. The domain information of R^0 (i.e., its vertices' coordinate) and the hyperline h^0 used in R^0 partition are stored as a root node of BSP tree. Then, R^0 is divided by the selected h^0 into the positive and negative planes R^{0+} and R^{0-}, reflecting whether the dot product of each vertex point p of R^0 with h^0 is negative or positive. The resulting two half planes can then be similarly partitioned by additional hyperline, h^1. This recursive space partitioning continues until there is no remaining hyperline in its sequential list and the final unpartitioned region, called the BUS is represented at the lowest level of hierarchy in the BSP tree, i.e., not having any child internal node. This BSP tree makes the encoding of images represented by a more compact description of tree structure in terms of the number of polygons, BUSes.

Once the BSP tree is finally generated, the leaf-nodes representing unpartitioned regions (BUSes) are collected from the BSP tree in order to construct the BUS space. Then, a BUS adjacency graph is computed by the BUSes collected to efficiently perform a high-level of building extraction process. The BUS adjacent graph $G=(V,E)$ is an algebraic data structure which consists of a set of nodes $V=\{v_1, v_2, ..., v_n\}$ and a set of arcs $E=\{e_1, e_2, ..., e_m\}$. Each node v_k represent a BUS polygon B_k, where the polygon vertices and statistical information over member pixels within B_k (i.e., intensity mean μ_k and variance σ_k) are recorded. Each arc, e_l, represents an unordered neighbouring relation of nodes $\{v_i, v_j\}$. This adjacent relation, e_l, is simply obtained by querying a BUS, B_j, which shares two consecutive vertices of B_i in order. A similar data structure, known as PAR (Polygon Adjacency Relationship), was used in Ameri & Fritsch (1999) for reconstructing the building roof structures. However, the PAR is constructed

through rather complicated computation of the Voronoi diagram based on distance transformation. In contrast, the computation of BUS adjacent graph is much simpler than the PAR because the image segmentation by the BUS space is completed without having the gaps between segments and a topological query becomes much easier in a polygonal space such as the BUS space.

Figure 5-9 illustrates a result of the generation of BUS space. Here, the median value of the pixel intensities within each unpartitioned polygon is used as the surface brightness model. This was obtained using a texture mapping technique from integer image space to real convex polygon space. As can be shown in this figure, the abstract level of object representation can be controlled by choosing a suitable scale space in the BSP tree. That is, the depth of the BSP tree can be controlled by determining how many hyperlines are used in the partitioning process. When this scale factor, l_{pc} is defined as a ratio of the hyperlines used in the partitioning over the total number of hyperlines, a multi-scale representation of the building in the BSP tree can be changed by selecting different l_{pc}. Also, the result shows how the building roof, having complex superstructures and heterogeneous radiometric characteristics, can be reconstructed by generating the BUS space, where geometric properties of building boundaries are well preserved to be suitable for a further building description process, and a lot of noisy contents are removed.



(a) sample image (b) l_{pc}=0.3 (c) l_{pc}=0.6 (d) l_{pc}=1

Figure 5-9. Results represented by multi-scale space of BSP tree

5.4.4 BUS grouping

When entire lines (i.e., l_{pc}=1) are used for creating a BUS space by the BSP method, each BUS can be considered as an over-segmented sub-region in a conventional region growing algorithm as can be seen in figure 5-8(d); too many polygons are generated to reconstruct building outlines. This is caused by the fact that spurious lines besides building boundary lines are involved in the generation of BUS space. A BUS grouping process starts from this over-segmented BUS space, and eliminates spurious lines by merging statistically similar BUSes. As a result, only BUS polygons, which have a line segment corresponding to building boundaries, are remained. Finally, building outlines are reconstructed by integrating those BUSes.

Now suppose that the entire image domain I is segmented into M piecewise homogeneous BUSes $\{B_i\}_{i=1}^{M}$, i.e. $I = \cup_i^M B_i$, $B_i \cap B_j = 0$, if $i \neq j$. The BUS grouping process starts from a seeded BUS B_0, which is automatically selected as the polygon containing a seed point selected by the user. A statistical probability of $P_0\left(I;\alpha_0\left(\mu_0,\sigma_0\right)\right)$ is measured over the member pixels of B_0, where α_0 is the parameters of the gray-level distribution and $\left(\mu_0,\sigma_0\right)$ are sample mean and sample standard deviation respectively. It is assumed that B_0 is constructed around a centre of targeted building roof by the user's visual inspection, and α_0 is consistent with the overall statistical characteristic of the target building roof.

However, the size of B_0 is variable, obtaining more reliable α_0 is necessary. To this end, P_0 is maximized as B_0 is explicitly grown. That is, the mean differences between B_0 and its neighbouring BUSes, B_N, are measured, and a neighbouring BUS B_N^i with the minimum mean difference is merged with B_0 if it is less than pre-defined threshold for the homogeneous predicate. This process proceeds iteratively until either the size of merged B_0 is less than certain threshold or a maximum iteration number is reached. At each iteration step, the initial homogeneity threshold linearly increases according to the iteration number. In this way, the initial seeded BUS, B_0, is explicitly grown to be B_e, and P_0 is maximized to be $P_e\left(I;\alpha_e\left(\mu_e,\sigma_e\right)\right)$.

Now, α_e is fixed during the BUS grouping process as representing overall intensity distribution of the target building roof. Once a homogeneity predicate is defined, B_e searches for its neighbours in the BUS adjacent graph and the statistical distribution, $\left\{\alpha_N^i\left(\mu_N^i,\sigma_N^i\right)\right\}$, of neighbouring BUSes, $\left\{B_N^i\right\}$, are computed. A neighbouring BUS, B_N^i, is merged with B_e if a statistical distance between α_N^i and α_e is satisfied with the predefined homogeneity predicate. Note that the homogeneous predicate is differently applied to each building depending on a "coarseness" of the building roof (i.e., homogeneous or heterogeneous building roof). This binary classification of the building roof property is determined according to σ_e. The aforementioned grouping process is described as follows:

$$\left|\mu_i-\mu_e\right|\le h_\alpha(\sigma_e)\times\mu_e \quad and \quad \left|\sigma_i-\sigma_e\right|\le h_\beta(\sigma_e)\times\sigma_e \tag{5-4}$$

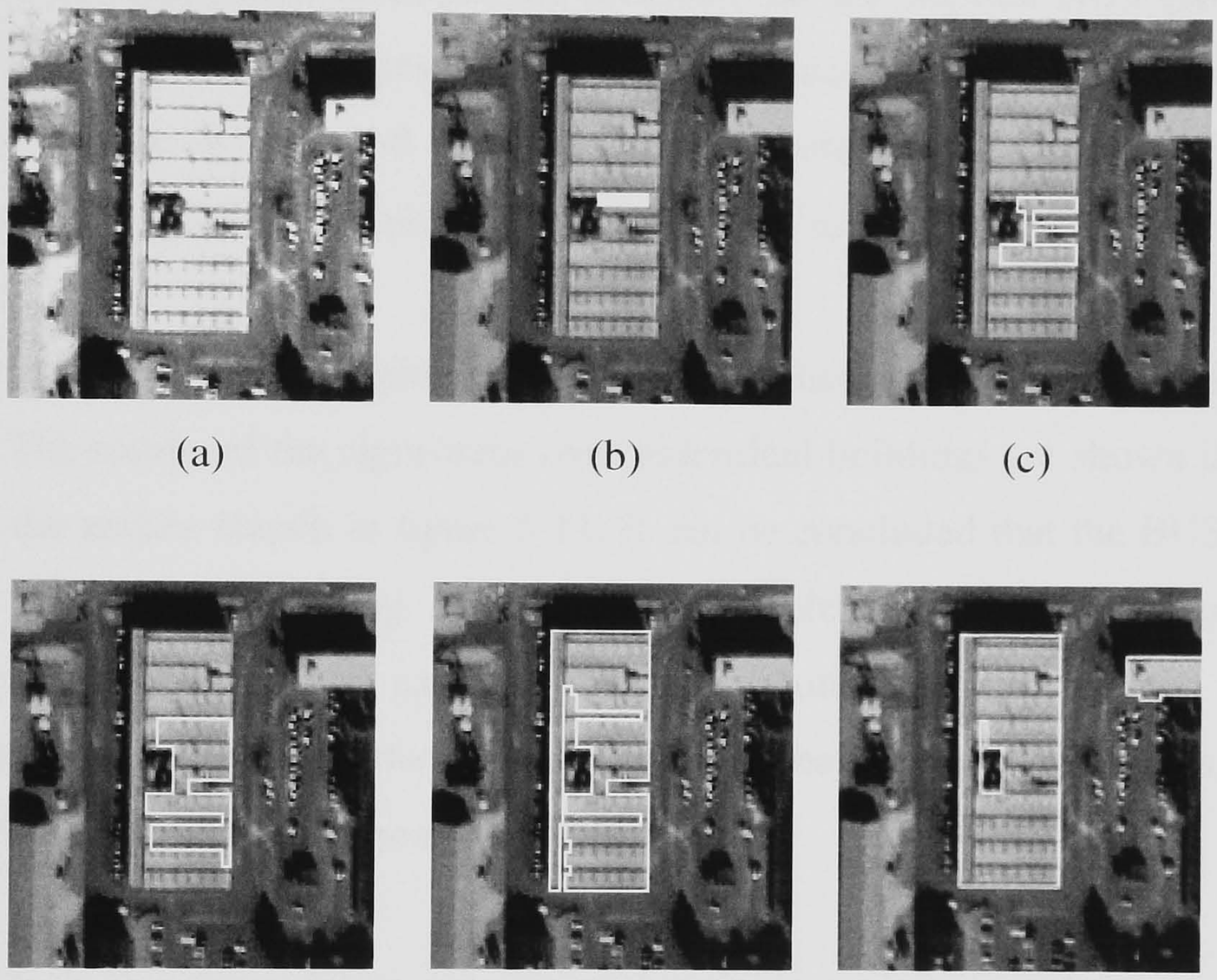
$$\begin{cases} h_\alpha(\sigma_e)=1 \ and \ h_\beta(\sigma_e)=1 & if\ \sigma_e\le 10 \text{ (homogeneous roof)} \\ h_\alpha(\sigma_e)=2 \ and \ h_\beta(\sigma_e)=2 & if\ \sigma_e>10 \text{ (heterogeneous roof)} \end{cases}$$

where μ_i and σ_i are the mean and standard deviation of the gray-level distribution in the compared neighbouring BUS, B_i, and h_α and h_β are weighting values for homogeneous predicates assigned according to the "coarseness" of building roof property. This BUS grouping process is terminated when there is no merged neighbouring BUS satisfying the implicit homogeneity predicate of Eq. (5-4) in the BUS adjacent graph.

5.5 Experimental results and discussion

This technique was tested on IKONOS monocular panchromatic imagery of the area around Farnborough in the United Kingdom. The user manually localized every target building, which was bounded by a rectangle with the size of 70x70 (m^2). Within each bounding box, linear features larger than 5-metre length were extracted by the Burns algorithm described in §5.4.1. Using those lines, a set of dominant building orientations was analyzed by analytical Fourier transformation so that only lines following dominant orientations remain and their noisy geometries are regularized as discussed in §5.4.2. In

the current experiment, it is assumed that the building structure can be reconstructed by only two major directionalities, and minor orientations were ignored. Then, the BUS space was generated by the BSP tree algorithm, where the partitioning function was used as described in §5.4.3, and a scale factor l_{pc} is selected as 1 for generating the BUS space.



(a) (b) (c)

(d) (e) (f)

Figure 5-10. Illustration of BUS grouping process

Finally, the BUS grouping process was applied to the generated BUS space in order to extract building boundaries. Figure 5-10 illustrates an example how the BUS grouping works for reconstructing building outlines. A BUS space was generated as shown in figure 5-9(d), and a seed BUS was selected by the user as shown in figure 5-10(b). This seed BUS was explicitly grown until its size increases up to the maximum size of 500 pixels. To this end, the sample mean distance, i.e., 20, was used as the homogeneity criterion for merging similar neighbouring BUSes. However, if current neighbours cannot be merged with that criterion, and the total size of the grown BUS has not yet reached to the maximum yet, the homogeneity criterion was linearly increased by 50%,

and the explicit growing process started again until the conditions of terminating the process are satisfied.

Figure 5-10(c) shows a final result of the explicit BUS growing process. A statistical gray-level distribution of $\alpha_e(\mu_e, \sigma_e)$ was computed in the grown BUS, by which the property of the building roof was determined whether it is homogeneous or heterogeneous. Finally, the homogeneity predicate for the implicit BUS grouping was newly defined, and the BUS grouping process continued until no BUS can be merged by Eq. 5-4. Figure 5-10(d) and (e) shows intermediate results of the BUS grouping process, and finally, a building outline can be obtained as in figure 5-10(f).

The developed technique mentioned above was applied to several different types of buildings. The results of the algorithms over individual buildings are shown in figure 5-11. From the results shown in figure 5-11, it can be concluded that the BUS grouping algorithm can extract building outlines with complex type of roof structure and microstructures. In particular, extracted building boundaries were clearly delineated with rectilinear lines though the contrast between the roof and ground is poor and complex textures exist around boundary regions.

This is difficult to achieve when conventional region growing algorithms are applied to the building extraction problem. This advantage is due to the fact that; the BUS is aerial features converted from linear features, whose lateral lines well respond to the image discontinuities; on the other hand, the facet of the BUS consists of member pixels, larger than one pixel, which helps the BUS grouping process measure more reliable statistics and makes the process insensitive to irregular textures of building rooftops. Moreover, the result was generated without employing specific rules for feature grouping, i.e., collinearity, orthogonality, and parallelism such as in the perceptual organization method. Thus, the BUS space algorithm can avoid using an "accidental cause" for constructing building hypothesis under the circumstance of the significant lack of line cues. The complexity of building structure is restricted to two major orientations in the current implementation. However, in theory, the BUS algorithm can cope with more than two directionalities for extracting building outlines if more reliable cues for verifying a building BUS can be supported in the BUS grouping process besides the gray-level homogeneity.

(a) (b) (c)

Figure 5-11. Building boundaries reconstructed by BUS grouping process; (a)-(c) are targeted buildings localized from IKONOS imagery of Farnborough area in U.K.

However, a drawback to the suggested algorithm occurred in the case that significant linear feature cannot be recovered through the BUS space generation so that the overall building outlines are extended to the ground due to the lack of lines. This problem can be seen in figure 5-12 (see the buildings located around the centre of Farnborough sub-scene). It can be seen that the global orientation analysis of building cluster is not enough to infer the totally missed lines by the Burns algorithm. To work out this problem, simple building models such as parallelism or orthogonality could improve the process to recover those missing lines if the models are involved in extracting the linear features. In the next chapter, the balance between data-driven and model-driven feature extraction will be discussed.

Another problem was inherited from the nature of the BUS grouping process; generic building rooftops cannot be restricted to certain range of gray-level homogeneity, and the algorithm can be applied to a limited type of buildings. Also, the grouping mechanism still relies on a local processing to group similar BUSes and hence, the process can rarely be proven to converge to a global solution, i.e., a ground truth model of building outline. In addition, like most region growing algorithms, the BUS grouping

process requires a post-processing algorithm to regularize small noisy-artifacts, i.e., intrusion or extrusion of the building boundaries. However, it can be achieved more easily by a morphological filter suggested by Baillard et al. (1999) since the resulting artifacts have regular shapes and their directionalities are known. In the next chapter, the aforementioned drawbacks will be overcome by combining 3D information with the BUSes, where the height information of underlying terrain surface will be used to determine whether a BUS actually belongs to the rooftops or not.

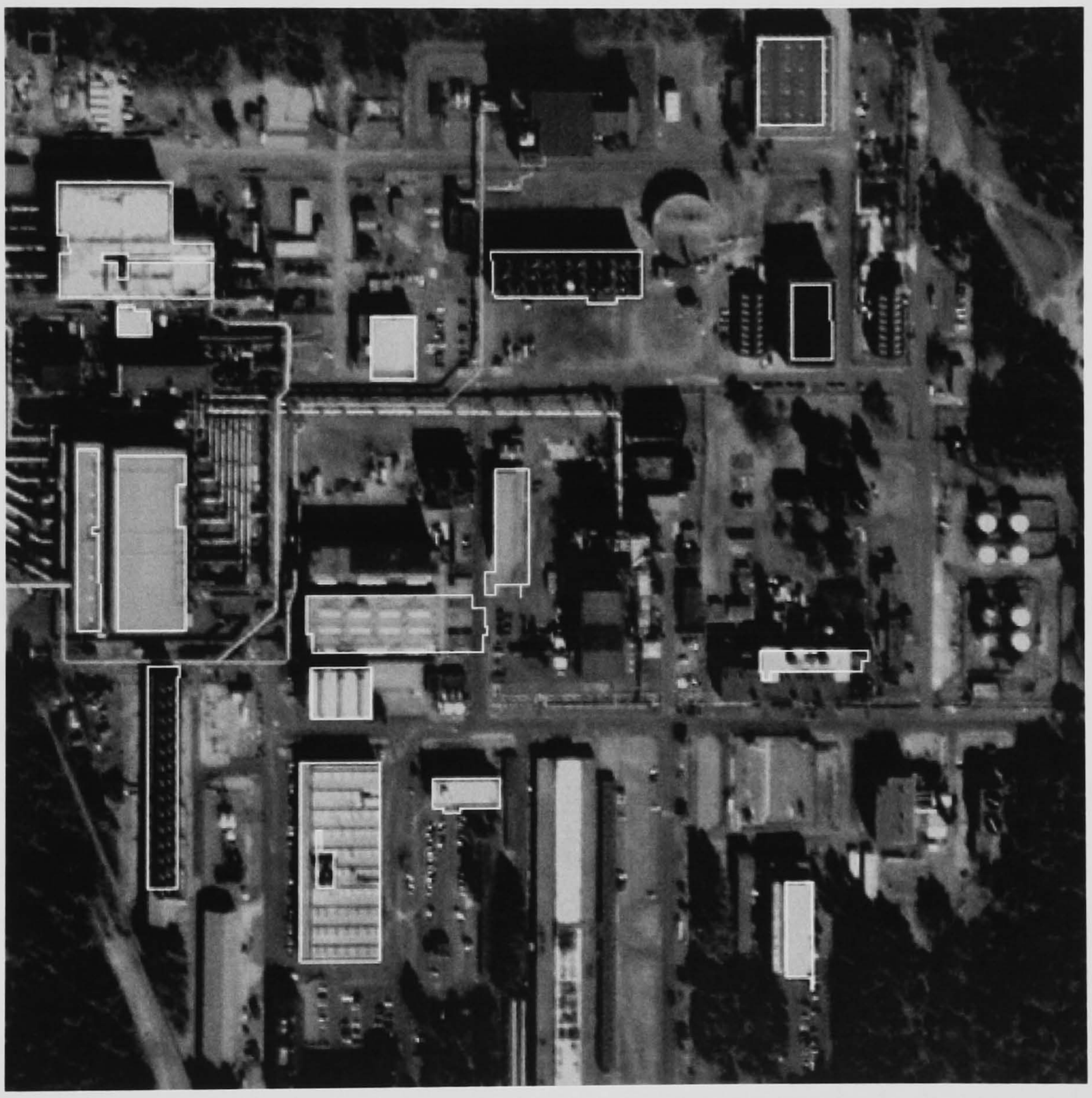


Figure 5-12. Final result of building boundary reconstruction by BUS algorithm

5.6 Summary

The problems of accurately describing building shapes from monocular imagery are typically under-constrained. Information such as absolute scale and depth is lost when the scene is projected onto an image plane. Even the same scene may give rise to very different images depending on the viewpoint and illumination conditions. In these circumstances, inferring an accurate object description from a large number of hypotheses is a very difficult task. As a preliminary study for developing a building extraction system, this chapter has discussed difficulties in delineating building boundaries from monocular imagery and exploited new techniques to overcome them.

In §5.2, the perceptual organization method has been reviewed as a possible solution to extracting building boundaries from remotely sensed data. Resolution effects on the building extraction system using 1-metre panchromatic imagery of IKONOS satellite have been discussed as a major bottleneck in adopting the method. Due to the low signal-to-noise ratio and limited resolving power of IKONOS imagery, the extracted geometric cues are often highly fragmented or completely missed, which makes it difficult to generate building hypotheses and to verify them. Moreover, those factors degrade the reliability of feature grouping rules used for recovering the insufficient cues. Region-growing technique has also been discussed as an alternative solution to the perceptual organization method. The technique is better in noisy images, and it produces a complete description of object boundaries. However, the region-growing technique frequently produces building boundaries with irregular geometric shape, which do not coincide with real edges.

In §5.3 and §5.4, a few new ideas have been introduced to overcome the difficulties posed by the perceptual organization and region-growing method. A generic building shape was represented as a mosaic of convex polygons with similar gray level. Compared to parametric models, this is suitable for describing complex shapes of buildings. A global analysis of dominant building orientation was suggested as useful knowledge that helps the process reduce the complexity of the building extraction task. A set of major directionality forming a building shape is analyzed by an analytic function of Fourier transformation. Based on this information, only straight linear

features, which are relevant to building outlines, are retained, and their geometric parameters perturbed by the noise are regularized.

A BUS (Building Unit Shape) organization method was introduced as a middle-level of feature inferring and grouping solutions, which can recover highly fragmented or completely missed lines, and finally reconstruct building outlines. A BUS space represents a complete description of image space with a set of convex polygons, which is generated by the BSP (Binary Space Partitioning) method. The BSP "optimally" segments the entire image into polygons by recursive intersections of lines extracted. Each polygon, called BUS, represents a piecewise homogeneous aerial feature encompassed by discontinuities. Similar to the region-growing technique, a BUS grouping method was suggested to eliminate spurious lines and thus to retain only building boundaries by merging statistically similar BUSes.

In §5.5, the technique proposed in this chapter was applied to a sub-scene of IKONOS imagery covering the Farnborough area. The result has shown that the method can extract building outlines from monocular IKONOS imagery and overcome the shortcomings of conventional area-based segmentation. The advantage of the method mostly comes from its flexibility. In theory, the polygonal image segmentation provides a framework to delineate a variety of building shapes (except curvature-like shape). However, the results have also shown some drawbacks of the technique, which must be resolved in future works. These problems can be described as follows:

1. The method requires that individual buildings of interest must be manually detected and localized by the users.
2. The representation of building objects is restricted to a certain range of photometric homogeneity, even though it provides a framework to give a high-level of geometric freedom for describing building outlines.
3. The method suffers from the difficulty in producing a complete description of building outlines in case of significant parts of building outlines being completely missed. The global orientation analysis of building clusters to overcome the aforementioned problem is not enough to infer the totally missed lines.

All the problems mentioned above indicate that the proposed technique must be improved to automate the building detection procedure and to relax "hard-constraints," such as photometric similarity in order to more generally represent the building object. In addition, a technique to recover completely missed line cues from monocular IKONOS imagery is required. In the next chapter, the aforementioned problems will be mostly resolved by combining 3D information provided by LIDAR with the BUS organization method presented in this chapter.

CHAPTER 6

BUILDING EXTRACTION USING IKONOS IMAGERY AND LIDAR DATA

6.1 Introduction

The previous chapter described a semi-automatic technique to delineate the building boundaries using a monocular IKONOS imagery. The proposed method was based on the building unit shape (BUS) organization technique, which recursively obtains the building polygons by data-driven linear features. The BUS method showed a promising aspect to achieve the building boundary representation with minimal use of feature grouping rules. There are, however, two major problems to degrade the performance of the algorithm. First, since linear features are extracted from a monocular image, it is difficult to recognize which features correspond to the building boundaries. In the previous chapter, it was assumed that this building detection can be given by an end user. For a building extraction system to be useful, however, the building detection needs to be automated. Second, although the BUS method provides a framework to recover fragmented or missed linear cues, the algorithm may fail when these are significantly lost. This problem will be obvious when only a monocular image of urban area is used. Because of these reasons noted above, rather strong building constraints were used, in which building orientations and material properties of rooftops are restricted. There is, however, a much wider range of building shapes in an urban environment.

This chapter aims to develop a new building extraction system to overcome the drawbacks mentioned above. Some improvements are achieved through a fusion

principle, which combines 3D information with monoscopic linear cues. As discussed in §2.5, 3D information has been recognized as an important cue for building extraction. In tradition, 3D information has been acquired by using multiple optical images. Many authors have emphasized the use of 3D cues, where 3D corners (Fischer et al., 1998), 3D lines (Baillard et al., 1999), and 3D planar polygons (Ameri & Fritsch, 2000) were used as primary cues for building extraction. Relying on 3D information generated from multiple images, the building extraction becomes much more robust compared to the 2D feature analysis method. Due to the rich texture of urban scenes, an enormous number of distracting features are usually extracted. Amongst them, only features belonging to buildings can be efficiently classified by the support of 3D information, based upon the assumption that the height of building objects is higher than the ground surface. As the distracting cues are excluded in advance, a size of building hypotheses generated can greatly decrease, and thus, the complexity of their verification process is reduced. On the other hand, 3D cues provide multiple views of individual objects since they are usually extracted by using two or more images. A cue deficiency, which is caused by low contrast, occlusion and shadow effect, can be recovered to some extent by multiple views acquired at different dates with various light conditions. Finally, the 3D analysis of primary cues makes it useful to understand the building structure as the parameters of building models used are estimated by data-driven 3D cues. It can hardly be achieved by 2D feature analysis.

However, 3D cue extraction using optical images has some drawbacks. It requires a very good combination of multiple images to recover "novel" views of individual buildings, covering all the parts of objects. In addition, it is not the usual case to obtain 6 or more images such as the work of Baillard & Zisserman (2000), even stereo images for some applications. Moreover, the technique of extracting 3D information from optical images has not matured yet due to the difficulty of finding homologous points across multiple images. As a result, the cue density obtained varies depending on the scene contents and image quality. Also, the accuracy of 3D information acquired from optical images is not enough for some applications. Indeed, the results of the mid-level processes in extracting and structuring 3D primitives is highly dependent on the quality of the utilized 3D cues, which in the worst case leads to partially or completely wrong descriptions of the buildings.

Thus, other data sources have been exploited to compensate for the disadvantages of optical imagery. In this area, LIDAR data has been exploited as an attractive alternative to optical photography due to high vertical accuracy and high point density. For building extraction, LIDAR data has been used to reconstruct various types of building shape; parametric model (Mass & Vosselman, 1999; Wang & Schenk, 2000), prismatic model with flat roof (Weidner & Forstner, 1995) and polyhedral buildings with the restriction on building orientation (Vosselman, 1999). Although LIDAR data have some advantages of building detection and planar patch extraction compared to aerial photographs, there is, however, a significant drawback to delineate building boundaries with break lines, even with extremely high density of 7 points per square metre (Vosselman, 1999). Therefore, a fusion technique is recommended to combine the complementary nature of the two different data sources for building extraction (Schenk & Csatho, 2002).

The research described in this chapter aims to exploit the synergy of single IKONOS satellite imagery with 1-metre resolution, and LIDAR data with one point per 3-metre point spacing. In the suggested building extraction system, LIDAR measurements are used for providing high-quality 3D information of the terrain surface and non-terrain features, while IKONOS monocular imagery provides colour information and straight linear cues. Both data serve to sequentially detect and remove dominant urban features that are not relevant to buildings, thereby isolating individual buildings from their surroundings, and then building boundary lines are obtained. For recovering linear cues missed from IKONOS imagery, new linear cues are virtually generated by employing simple building models. A "good" balance between data-driven and model-driven cues is achieved as the BSP algorithm subsequently generates polygonal building parts according to their degrees of "level-of-detail" forming a building shape. Finally, a prismatic building model with polygonal shapes is obtained by merging verified building polygons.

6.2 Method Overview

The complexity of building extraction process can be reduced to a large degree if the process can be focused on a single building object. Therefore, it is desirable to localize individual buildings within the data domain used for building extraction. To reduce the scene complexity, a focusing strategy has been proposed by many researchers (Baillard & Maitre, 1999; Brunn & Weidner, 1997; Haala, 1994; Weidner & Forstner, 1995), in which individual buildings are separated from their surroundings. These approaches share a common strategy of hierarchical object detection, in which on-terrain features and off-terrain features are differentiated with the support of high-resolution height information and then, off-terrain features are further classified into tree and building object classes. As a final result, individual buildings are localized with spatial information bounding them, referred to as a building "blob".

For building detection, the proposed system adopts a similar approach mentioned above, in which feature classes are hierarchically categorized and a classification algorithm is selected according to the feature class targeted. Three class layers are prepared. At the coarsest level, on-terrain features and off-terrain features are differentiated according to whether those features are located either on the terrain surface or above the terrain surface. This classification is done as the RTF filter described in chapter 3 and chapter 4 is applied to a given LIDAR DSM. With the on-terrain points detected by the filter, a DTM is generated. Then, outlying points with a height less than a pre-defined threshold from the generated DTM are classified as "high-rise" features, otherwise as the "low-rise" ones. Since the "high-rise" feature class generally contains trees as well as buildings, vegetated points must be identified. At the finest level of classification, differentiating buildings from trees is made by the use of IKONOS multi-spectral bands. To this end, the normalized difference vegetation indices (NDVI) are computed by a combination of red and near-infrared channels of IKONOS. When LIDAR points are back-projected onto IKONOS image space, the "high-rise points" are associated with the NDVI information. Building points can be identified by removing vegetated points whose NDVI value is larger than a threshold value. Further processing allows the individual building objects to be bounded with a rectangle and these building "blob" polygons are then fed into the building extraction process with the final labels of LIDAR points.

The next step is to extract building boundaries from each building "blob" detected. The building extraction process has been developed based upon the BUS organization method, which has been exploited in the previous chapter. The strategy is, however, different by the contribution of LIDAR DSM, and the building constraints used are more relaxed, which allows the system to delineate more complex polygonal building shape. First, straight linear futures are extracted from IKONOS imagery, but at this time, the extraction of these "intensity" line cues is focused on building boundary lines by the support of LIDAR point labels. These "intensity" line cues will be used as a part of hyperlines for the generation of BUS space. The rest of hyperlines, referred as "virtual" line cues, are generated by employing simple building models. To this end, parallel lines and "U" structured lines are inferred over LIDAR space from each "intensity" lines extracted. When the generated "virtual" lines are back-projected onto IKONOS image space, their geometries are adjusted to fit to a maximal gradient magnitude.

Then, the hyperlines generated recursively partition a building blob polygon, which results in a BUS space with a set of convex polygons (i.e., BUSes). The process shares common methodology of BSP partitioning used in the previous chapter, but it has several different aspects. Firstly, the generation of BUS space is made over LIDAR space. Secondly, it adopts a level-of-detail (LoD) strategy. That is, the most "significant" part of building structure, which comprises only building label points with a maximum size, is obtained. Then a BUS with less "significant" part is subsequently generated. In this way, recursive building part segmentation continues until only distracting polygons are generated by hyperlines. The LoD strategy prevents erroneous "virtual" cues from being involved in building outline generation. Once the BUS space is generated, a heuristic filter verifies BUSes as building parts, which belongings to the building structure. Finally, the verified BUSes are aggregated, by which the boundary of polygonal building shape can be reconstructed. Figure 6-1 shows a systematic flow diagram of the proposed building extraction method.

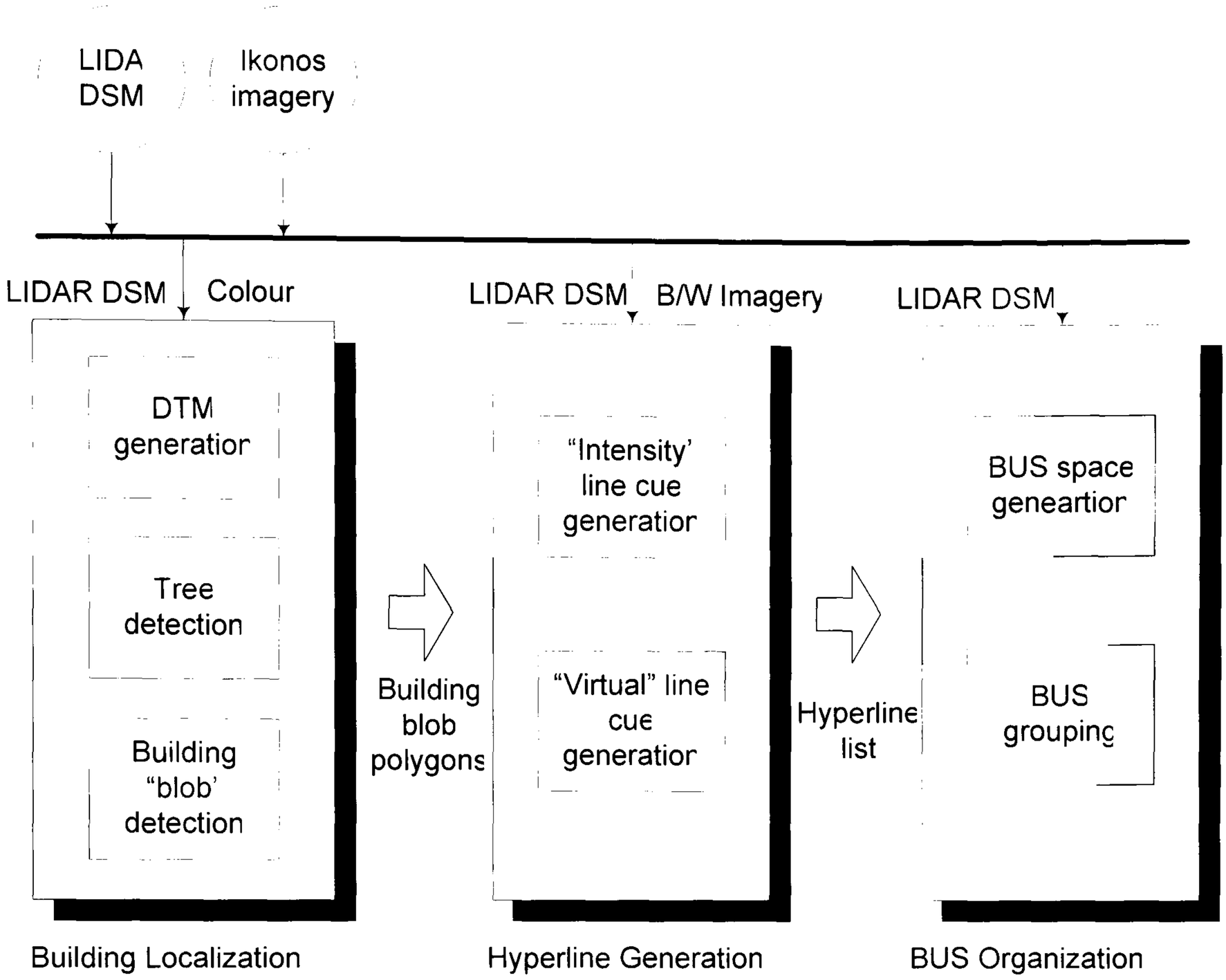


Figure 6-1. Process flow diagram of the suggested building extraction system.

6.3 Data set used

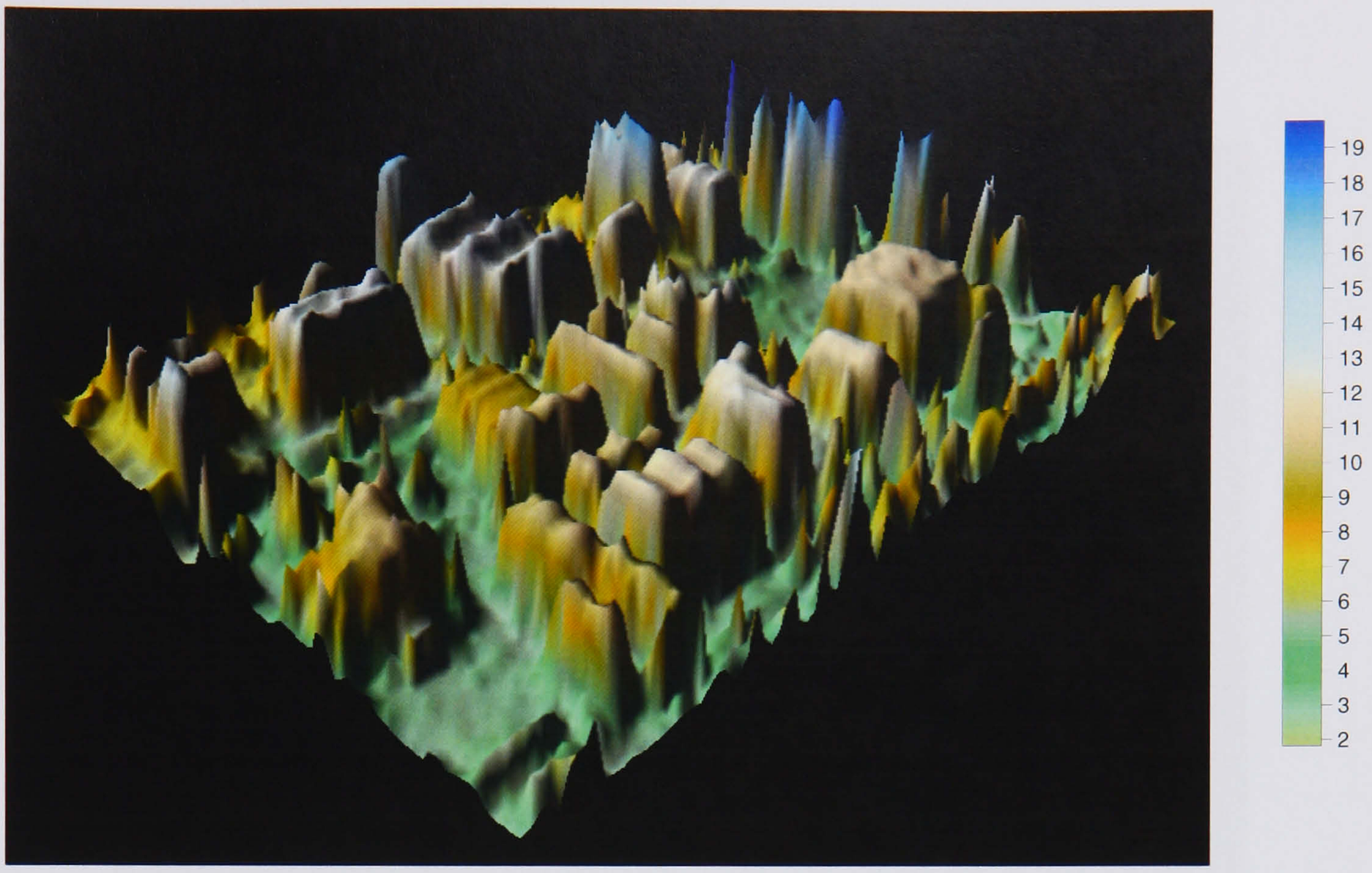
6.3.1 LIDAR DSM

For the current research a LIDAR DSM was provided by Infoterra Co., which covers a sub-site of Greenwich industrial area with the size of 305,523 m^2. The LIDAR DSM was acquired by the first pulse of OPTEC 1020 airborne laser sensor. The data has been converted from OSGB36 (plan) and OSD Newlyn (height) to UTM/WGS84. The characteristics of OPTEC 1020 laser sensor can be found in §4.2.2.1. The LIDAR DSM contains a total of 30,782 points, which corresponds to a point density of 0.1 (points/m^2), i.e., one point per 3.2 x 3.2 (m^2). The height of the study area varies from 1.4 m to 26.3 m. The data characteristics of the Greenwich LIDAR DSM are summarized in table 6-1.

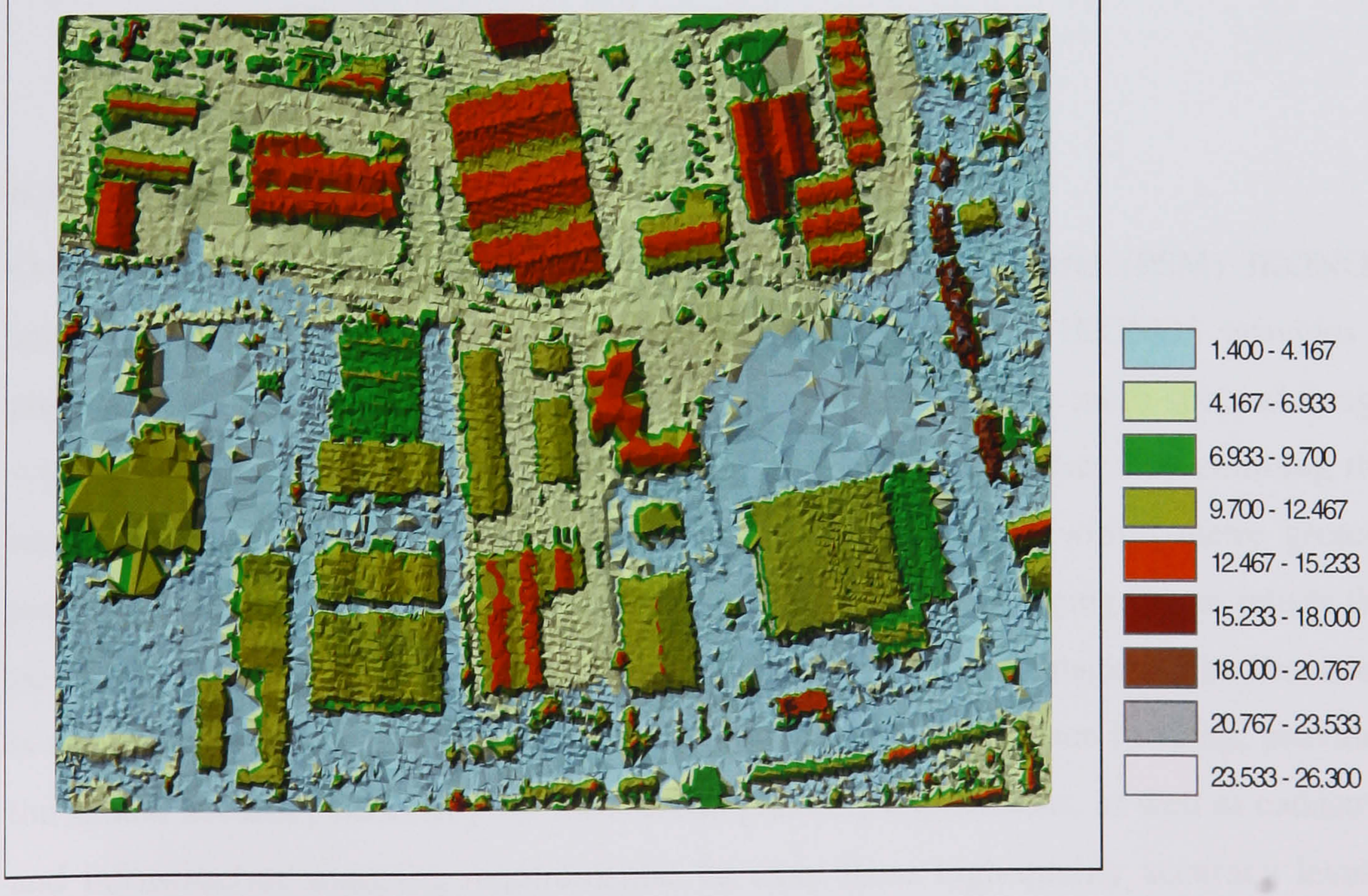
Table 6-1. Data characteristics of the Greenwich LIDAR DSM

Parameters	Values
Total number of LIDAR points	30,782 (points)
Point density	0.1 (points/m^2)
X range	293226.02 ~ 293904.96 (metre)
Y range	5708183.01 ~ 5708683.00 (metre)
Z range	1.4 ~ 26.3 (meter)

As can be seen in figure 6-2, the Greenwich LIDAR DSM shows a typical urban environment. The height of the terrain in this scene ranges from 1.4 m ~ 6 m. The terrain in the Northern part is higher than the Southern part, and the highest terrain height can be found in the North-West corner in figure 6-2. A number of industrial buildings with different sizes spread over the study area. The height of large buildings ranges from 7 m ~ 18 m. Some low objects with height of 5 m ~ 8 m can be found in the Greenwich dataset. In addition, a group of tall trees with various heights are located on the street. A row of terraced houses can be found at the lower right corner of the scene. In particular, figure 6-2 (b) shows the point density of the OPTEC 1020 LIDAR is not enough to properly represent the shape of those small houses. It also shows very irregular surface over some rooftops, though they are formed in planar roof surfaces. It suggests that the quality of the Greenwich LIDAR DSM is not good enough to cover the all the rooftops. This problem is highlighted when the Greenwich LIDAR measurements are overlaid with the IKONOS image in figure 6-3. Although the point density of the Greenwich data shows 0.1 points per square metres, it is much less over some building rooftops and the ground near buildings. This may be caused by certain factors such as poor weather condition, high absorption by specific material property, and occlusion effect. This problem may affect the building extraction process, which will be presented in this chapter. This will be discussed later in §6.6.



(a) LIDAR DSM in 3D



(b) LIDAR DSM in 2D

Figure 6-2. Greenwich LIDAR DSM

Figure 6-3. Greenwich LIDAR DSM overlaid with IKONOS image; red crosses represent LIDAR points

6.3.2 IKONOS PSM Image

Over the Greenwich study area, a "pan-sharpened" multi-spectral (PSM) IKONOS image is also provided by Infoterra Co. As described in §2.2, IKONOS provides a panchromatic image with a ground resolution of one metre and the multi-spectral image with a ground resolution of four-metres. The PSM image is produced by combing the multi-spectral data with the panchromatic data, and resampled with 1-metre ground pixel. This IKONOS PSM image was orthorectified by Space Imaging Co. to satisfy the positional accuracy (~1.9 metres) of Precision Product of Space Imaging. The Precision is the one of the most positionally accurate product offered by Space Imaging, provides the spatial accuracy necessary for most urban planning applications, as well as cadastral and infrastructure mapping requirements. To meet these high-quality accuracy levels, Space Imaging uses high-precision ground control and precise terrain models to create Precision products. Table 6-2 presents auxiliary information of IKONOS PSM image acquired over the Greenwich study area.

Table 6-2. IKONOS image metadata of Greenwich study area

Parameters	Values
Acquisition data	11-04-2000
Acquisition time	11:10 AM
Product level	Precision product
Projection	Universal Transverse Mercator (UTM)
UTM specific parameters	Hemisphere (North)
	Zone number (31)
	Datum (WGS84)
Bands	NIR
	Red
	Blue

Figure 6-4 shows the Greenwich IKONOS PSM image, in which the red channel is replaced with the near-infrared channel while the green channel as red channel respectively. The sub-scene image is 681 x 502 pixels with 1m resolution and therefore the dimension of the image is approximately 341,862 m^2. In this false colour composite the NIR reflectance is displayed in the red band.

Figure 6-4. Greenwich IKONOS PSM image

6.4 Building detection

This section describes a building detection method, which isolates individual buildings from their surroundings. The overall process is shown in figure 6-5. Three feature class layers are prepared, in which each layer is hierarchically produced. The next section describes the generation of the coarsest feature class layer. This process separates off-terrain features from the terrain surface reconstructed from LIDAR DSM. In §6.4.2, the off-terrain features labelled by the coarsest feature classification are then segmented further into either "low-rise" or "high-rise" feature class, and thus form the intermediate feature class layer. In §6.4.3, two different methods are introduced to identify vegetated points from "high-rise" features, where a local surface curvature and a colour information are investigated for characterizing vegetated features. With the remaining building points, a method to localize individual building objects is described in §6.4.4.

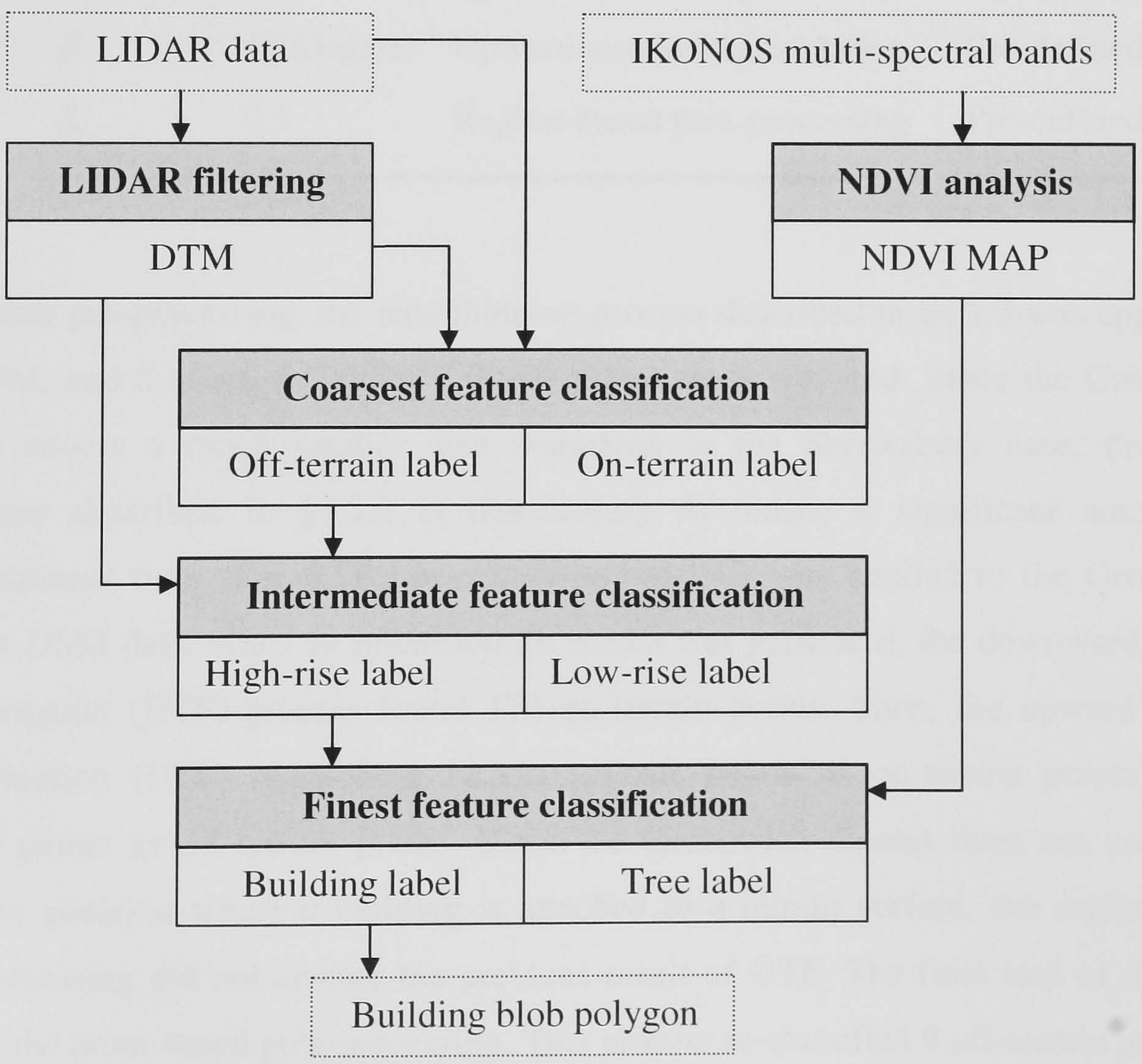


Figure 6-5. Suggested building detection process; the gray boxes represent the sub-processes involved; the white solid boxes describe the result generated by corresponding process; the dashed boxes are an input or output of the process

6.4.1 Coarsest feature classification

The first step of building detection process is to differentiate on-terrain points from off-terrain points in the Greenwich LIDAR DSM. To this end, the RTF filter described in §4.3 is applied to figure 6-2. Since the same sensor was used to acquire the Greenwich data, the same filtering parameters were selected as in the case of the Shrewsbury data described in §4.3. These parameters are summarized in table 6-3.

Table 6-3. RTF filtering parameters for the Greenwich LIDAR DSM

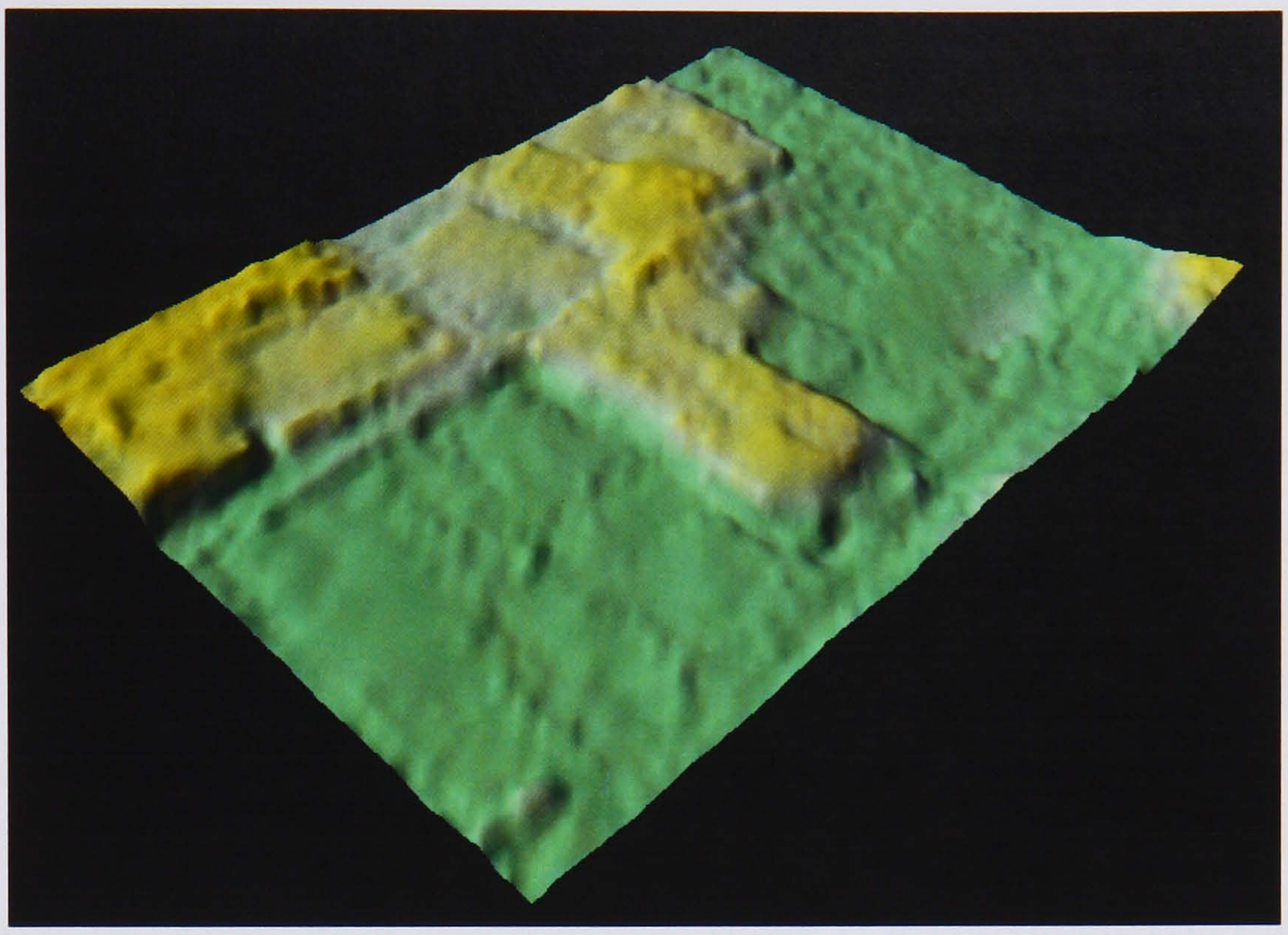
Parameter	Value	Process	Determination
κ	10 (meter)	Upward terrain fragmentation	Pre-defined
$\delta_{\min}$	0.5 (meter)	Upward terrain fragmentation	Pre-defined
$\delta_{\max}$	1 (meter)	Upward terrain fragmentation	Pre-defined
α	0.1	Upward terrain fragmentation	Pre-defined
β	45 (degree)	Upward terrain fragmentation	Pre-defined
δ_{PC}	0.3	Region-based post-processing	Pre-defined

For a data pre-processing, the data thinning process described in §4.2.3 was applied to the DSM, and 6 points of 30,782 LIDAR points were removed. Since the Greenwich dataset covers a much smaller area compared to the Shrewsbury case, the tiling procedure described in §4.2.3 is unnecessary to reduce a significant amount of computational time. The RTF filter presented in §4.3 was applied to the Greenwich LIDAR DSM data. After an initial terrain model was generated, the downward terrain fragmentation (DTF) process found 174 on-terrain points. Then, the upward terrain fragmentation (UTF) recognized 18,556 LIDAR points as on-terrain points, while 12,220 points as off-terrain points. Since the Greenwich dataset does not contain a complex situation where a building is attached to a terrain surface, the region-based post-processing did not change the previous result of UTF. The final step of the RTF filter is the point-based post-processing. This process re-classified 9 off-terrain points as on-terrain points and resulted in the detection of 18,565 on-terrain points and 12,211 off-terrain points. Table 6-4 shows the RTF filtering result over the Greenwich LIDAR DSM.

Table 6-4. RTF filtering result of the Greenwich LIDAR DSM

Processing step	Results (points)	
	on-terrain	off-terrain
Downward terrain fragmentation (DTF)	174	30602
Upward terrain fragmentation (UTF)	18556	12220
Region-based post-processing	18556	12220
Point-based post-processing	18565	12211
Total	30776	

Figure 6-6 (a) shows the terrain reconstructed by the RTF filter from the Greenwich LIDAR DSM, and figure 6-6 (b) overlays on-terrain points detected by the filter with the IKONOS image. As can be seen in this figure, the DTM is successfully reconstructed by the RTF filter, and most of on-terrain points are not located on the rooftops, despite the fact that underlying terrain surface consists of different terrain slopes and building objects with a range of different sizes are located.



(a) reconstructed DTM in 3D

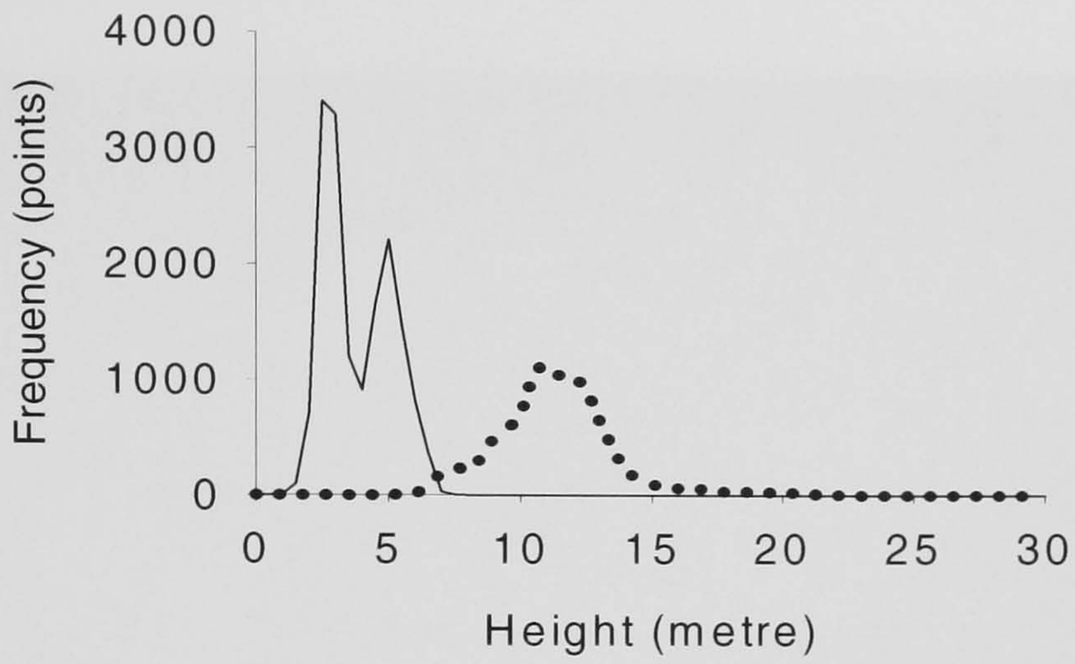
(b) on-terrain points overlaid with IKONOS image

Figure 6-6. DTM reconstruction result over Greenwich study area; red crosses represent LIDAR points labelled as on-terrain points

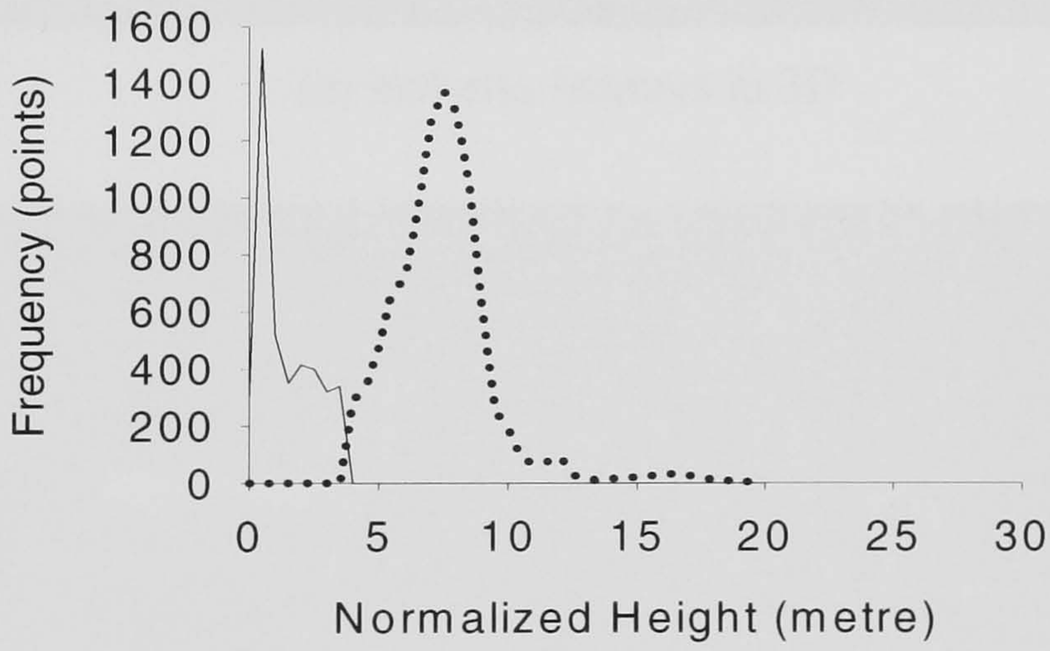
6.4.2 Intermediate feature classification

Once either an on-terrain or off-terrain label is assigned to every LIDAR point, off-terrain label points are re-classified into more detail feature classes, i.e., low-rise or high-rise feature class. The determination of this intermediate level of feature classification is made by comparing height attributes of off-terrain points with pre-specified height threshold.

Figure 6-7 (a) shows a height histogram of the result of the coarsest feature classification. As can be seen in this figure, the height histogram of on-terrain points obviously shows two maximum peaks. It suggests that the underlying terrain of the Greenwich dataset consists of two different terrain slopes. It is, however, difficult to distinguish between two classes (i.e., low-rise and high-rise classes) from the height histogram of off-terrain points since their heights are corrupted by the terrain surface. This problem can be solved by subtracting the height of underlying terrain surface from the one of off-terrain points. As shown in figure 6-7 (b), it becomes easier to distinguish between the low-rise and the high-rise class when the heights of off-terrain points are normalized by the terrain surface. Although a height threshold to classify off-terrain points into either low-rise or high-rise feature class is rather arbitrarily selected as 4-metre in this study, figure 6-7 (b) shows the selection of this height criterion can be automated by a statistical analysis of the normalized height histogram. Table 6-5 summarizes the result of the intermediate feature classification. Figure 6-8 and figure 6-9 shows final results of the intermediate level of feature classification, in which classified low-rise and high-rise points are located above the normalized terrain surface and overlaid with IKONOS imagery.



(a) on-terrain feature (solid line) vs. off-terrain feature (dash line)

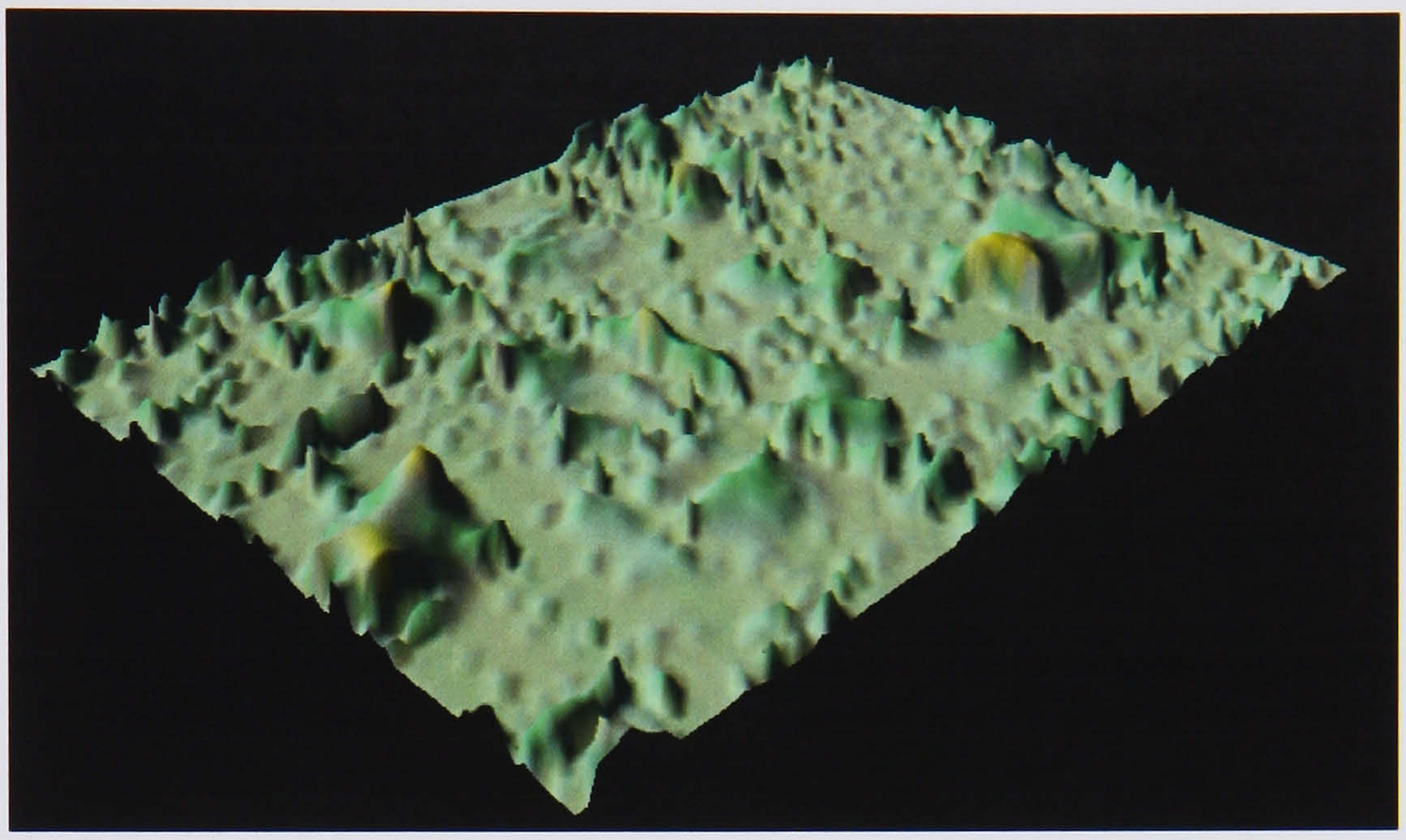


(b) low-rise feature (solid line) vs. high-rise feature (dash line)

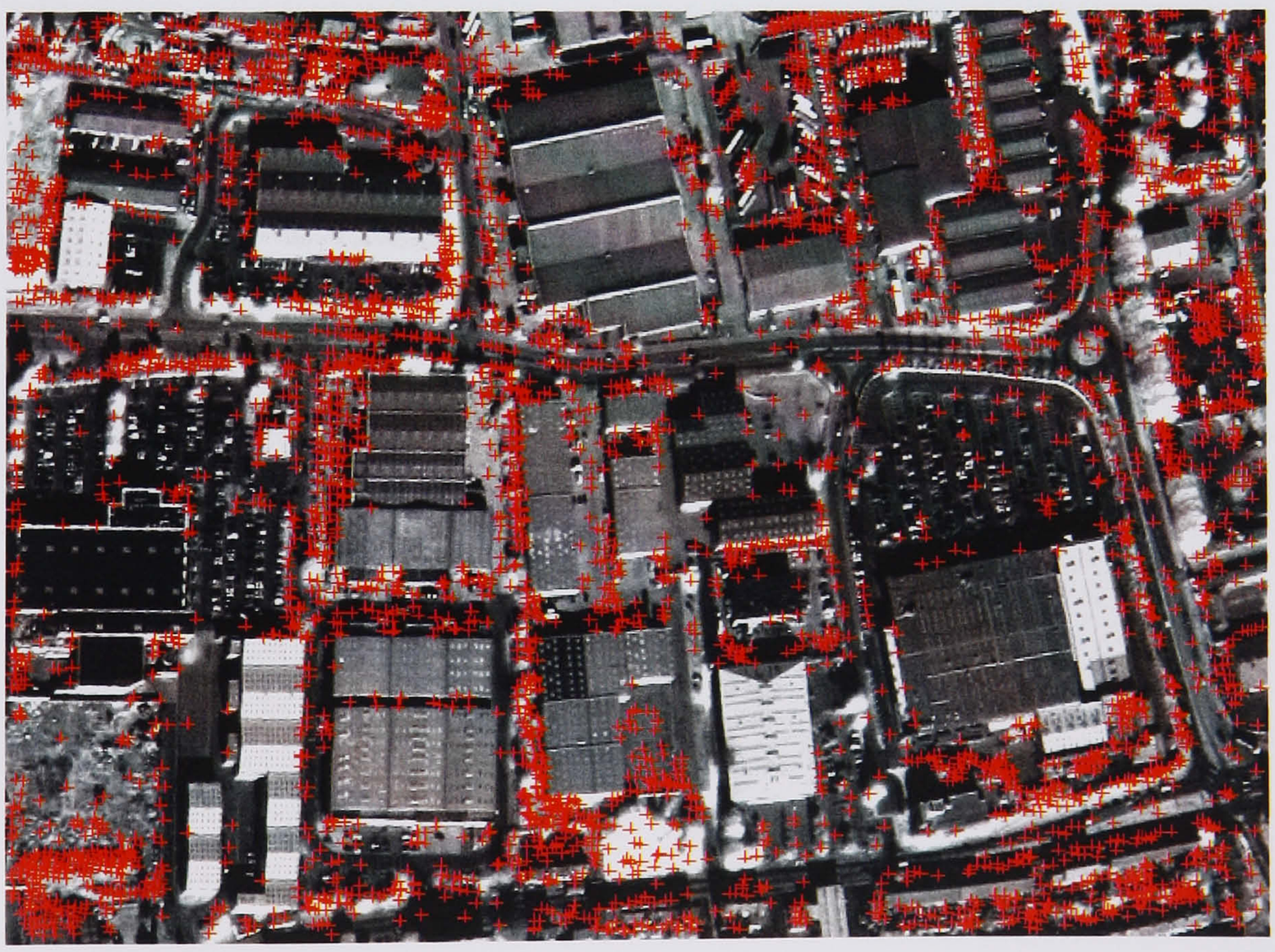
Figure 6-7. Coarsest and Intermediate feature classification results

Table 6-5. Intermediate feature classification result

Coarsest feature class	Intermediate feature class	Number of points
on-terrain label	-	18556
off-terrain label		12220
	low-rise	3586
	high-rise	8634
Total LIDAR points		30776

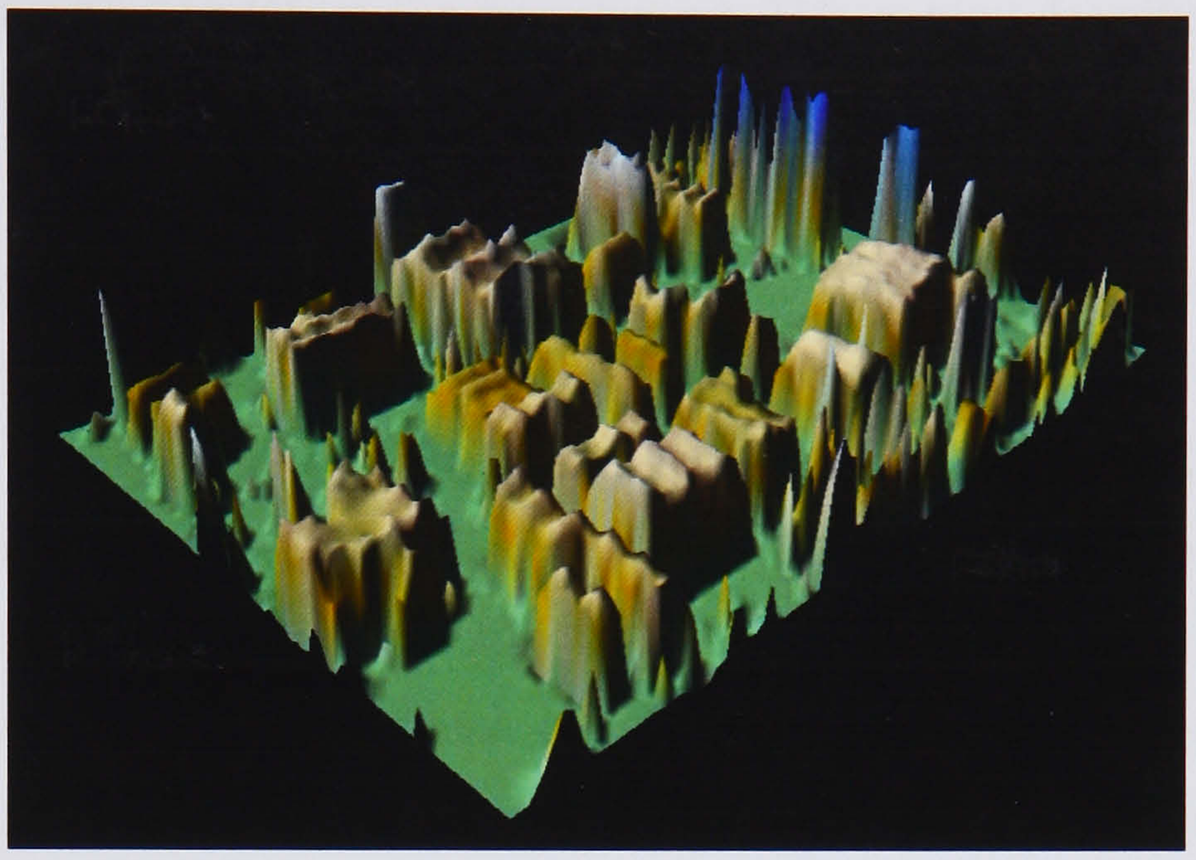


(a) low-rise features in 3D

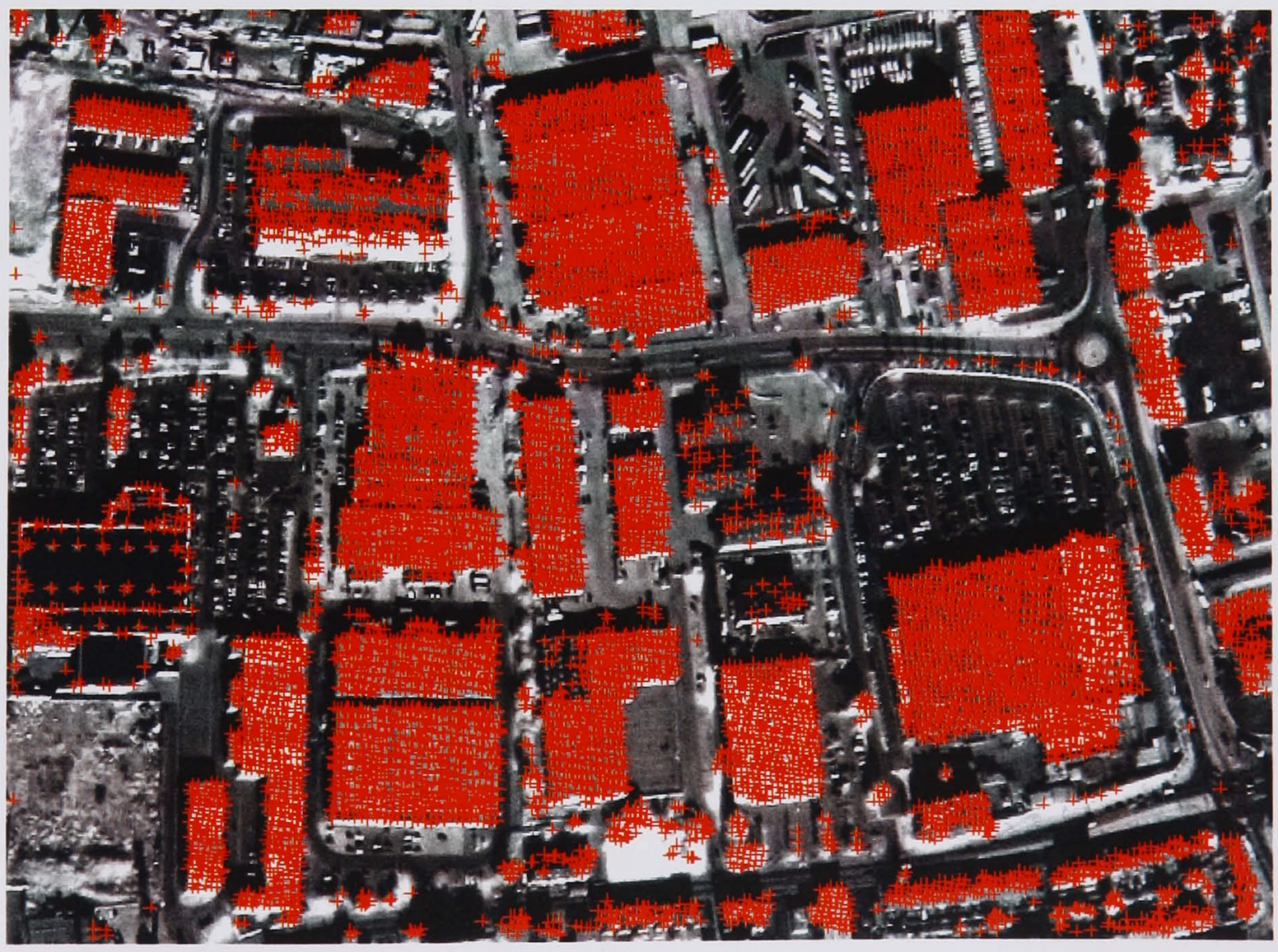


(b) low-rise label points overlaid with IKONOS image

Figure 6-8. Classification result of low-rise features over Greenwich study area; red crosses represent LIDAR points labelled as low-rise features



(a) high-rise features in 3D



(b) high-rise label points overlaid with IKONOS image

Figure 6-9. Classification result of high-rise features over Greenwich study area; red crosses represent LIDAR points labelled as high-rise features

6.4.3 Tree detection

The high-rise feature class is generally a mixture of tree and building objects. For building detection, the trees must be excluded from the high-rise features, by which the isolation of individual buildings can be achieved. This section describes two different methods to identify tree features. Compared to optical sensors, LIDAR measurements can represent a more accurate surface curvature of acquired objects. It is an obvious advantage to recognize certain object classes if their curvature characteristics are well constrained. In this section, a method is introduced to identify tree features by investigating a local surface curvature computed from irregularly spaced LIDAR points. The second method for tree detection is to use IKONOS multi-spectral information. Spectral information has been widely used as a primary cue for tree mapping application (Muller et al., 2001). Since surface material comprising an object responds differently to frequencies used, and produces its own characteristic reflectance, the spectral properties derived from multi-spectral information can be categorized into pre-specified object classes by traditional "per-pixel" classification technique. It can be used for the mapping of man-made objects as well as natural features in a complex urban scene.

6.4.3.1 Local slope variance analysis

For tree detection, it is assumed that the surface curvature of trees shows higher roughness than that of building rooftops since the tree object generally consists of "unstructured" parts. The surface roughness can be measured by a local slope variance, and thus, the surface curvature of trees is characterized as relatively high slope variance, while low over building rooftops. To compute a local slope variance, a TIN is constructed from irregularly spaced LIDAR points by Delaunay Triangulation method. Once a LIDAR point, s_i, with more than two connected points is found from the TIN, a slope, $\Theta(s_i, s_j)$, between $s_i(x_i\ y_i\ z_i)$ and its connected point $s_j(x_j\ y_j\ z_j)$ is given by

$$\Theta(s_i, s_j) = \tan^{-1}\left(\frac{z_j - z_i}{\sqrt{((x_j - x_i)^2 + (y_j - y_i)^2)}}\right) = \tan^{-1}\left(\frac{z_j - z_i}{d_u}\right) \quad (6\text{-}1)$$

where $\Theta(s_i, s_j)$ falls within a range of [-π/2, π/2]. In Eq. 6-1, an averaged point spacing of LIDAR data, d_u, is used to normalize the effect caused by irregular point spacing. Once all slopes are calculated between s_i and its neighbouring points, $N_i = \{s_j\}$ by Eq. 6-1, the min-max slopes of s_i are computed as follows:

$$\vartheta_{\min} = \arg\min_{\forall_{N_i}} \Theta(s_i, s_j);\ \vartheta_{\max} = \arg\max_{\forall_{N_i}} \Theta(s_i, s_j) \tag{6-2}$$

Then, a local slope variance (LSV), $\Delta\vartheta_i$ of s_i is obtained by adding up the min-max slopes as follows:

$$\Delta\vartheta_i = \left|\vartheta_{\min} + \vartheta_{\max}\right| \tag{6-3}$$

Figure 6-10 illustrates that a LSV, $\Delta\vartheta_0$, of a LIDAR point, s_0, is obtained from three different connecting situations. The largest slope variance of $\Delta\vartheta_0$ is made when s_0 is connected with s_1 and s_4. This implies that the absolute angle dispersion between the point and its neighbouring points is closer to π/2, and thus Eq. 6-3 computes higher LSV.

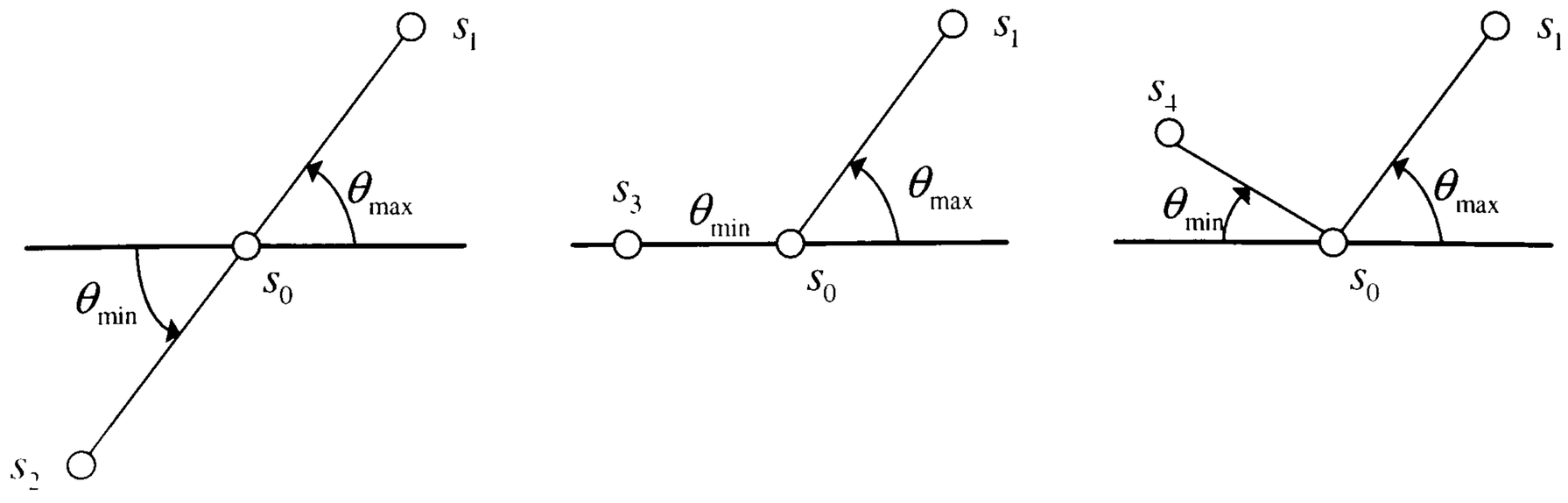
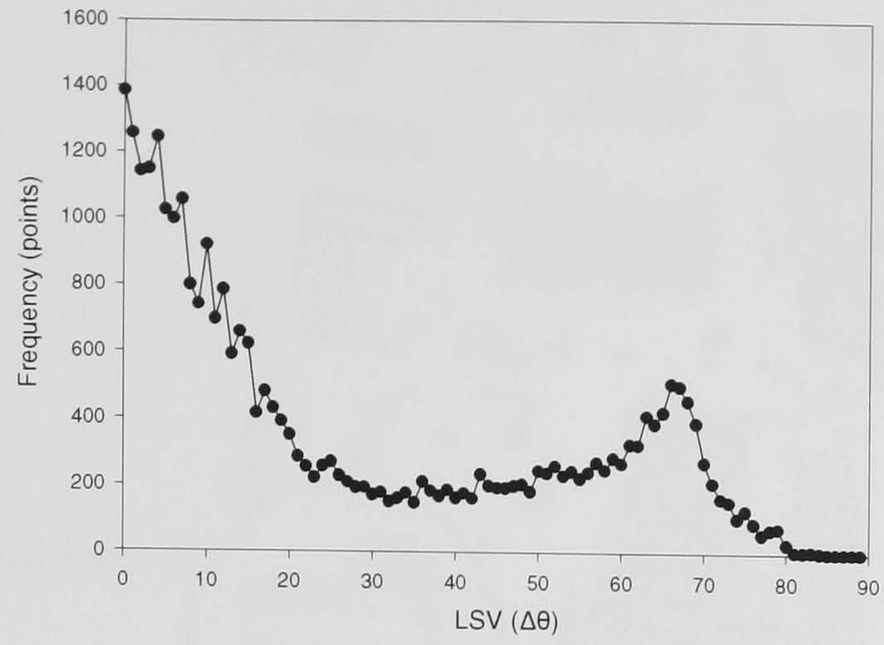


Figure 6-10. Illustration of a local slope variance computation

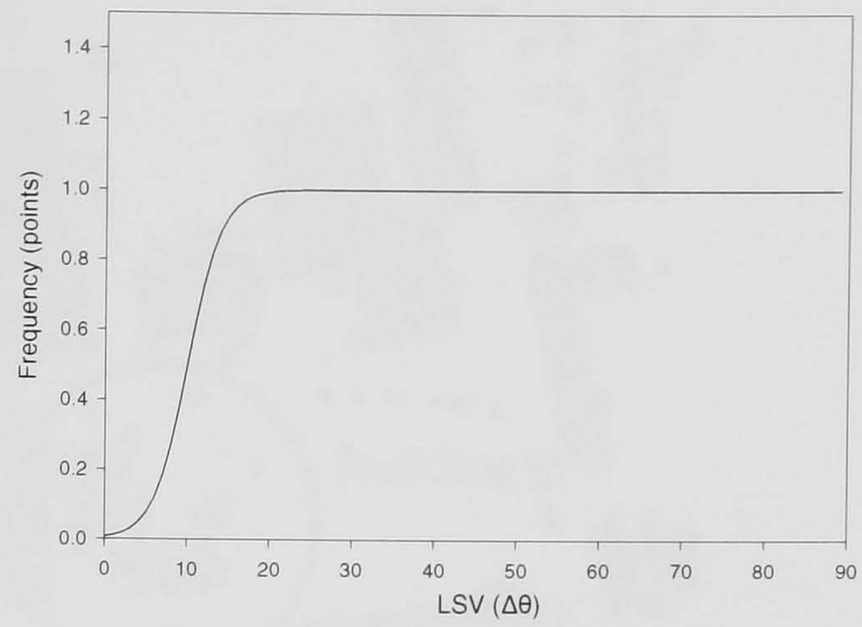
When Eq. 6-3 is applied to high-rise points classified at the intermediate feature classification stage, a histogram of computed LSV results in figure 6-11 (a). The result clearly shows a bimodal distribution, which is caused by the contribution of "structured" (i.e., building) and "unstructured" (i.e., tree) objects. Thus, the differentiation between two object classes can be achieved by a binarization of figure 6-11 (a). To select a threshold for the histogram binary classification, a weighted normalization function is applied to the computed slope variances as follows:

$$wnc_i = \frac{1}{1 + e^{-\alpha_w(\Delta\theta_i - \beta_w)}} \tag{6-4}$$

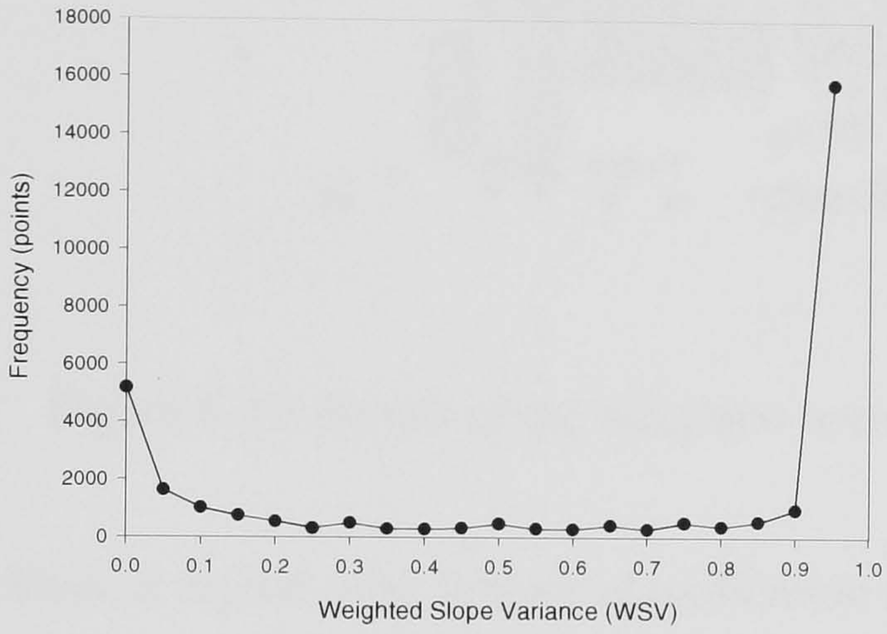
where $\Delta\vartheta_i$ and wnc_i is a slope variance and its weighted normalizing curvature for a point, s_i, respectively; α_w and β_w are parameters of a sigmoidal weighting function. α_w and β_w are selected considering a heuristic knowledge of the expected slope variance of the treetops. When α_w and β_w are selected as 0.5 and 10° respectively, figure 6-11 (b) shows the weighted normalized function of Eq. 6-4. The LSV larger than 20° is uniformly weighted with the highest value, while the weighting values of smoother slope variances less than 20° are exponentially decreased. When the local surface variance of figure 6-11 (a) is convolved with the weighted normalizing curvature of figure 6-11 (b), weighted slope variances (WSVs) are generated, which shows more biased bimodal distribution of local slope variances as shown in figure 6-11 (c).



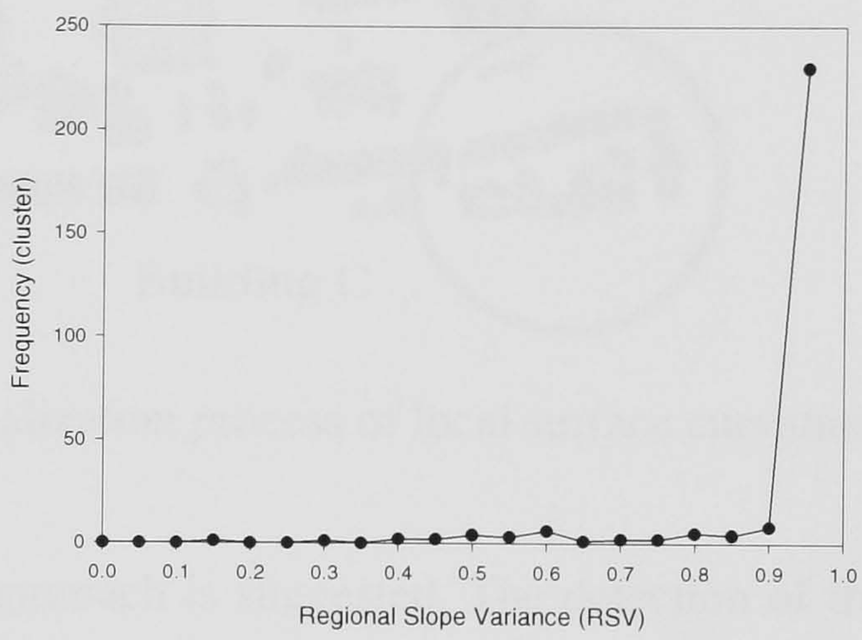
(a) LSV histogram



(b) normalized weighting function



(c) WSV histogram



(d) RSV histogram

Figure 6-11. Computation of weighted and regional slope variances of local surface curvature

Figure 6-12 shows a result when the WSVs are colour-coded over high-rise points. As can be seen in this figure, the differentiation between building and tree classes seems plausible by WSV. Most of LIDAR points with high values ("dark" dot) can be found over the trees and around building boundaries, while there are many low values ("light" dot) inside buildings. However, the result also illustrates a problem, in which highly weighted slope variances can be frequently observed inside the building roofs. This problem is caused by small roof constructions (e.g., chimneys, elevator equipment etc.) and intersections of complex building roof structures, which produce high change rate of slope angles. In this case, a point-wise classification removes trees as well as points located over building rooftops.

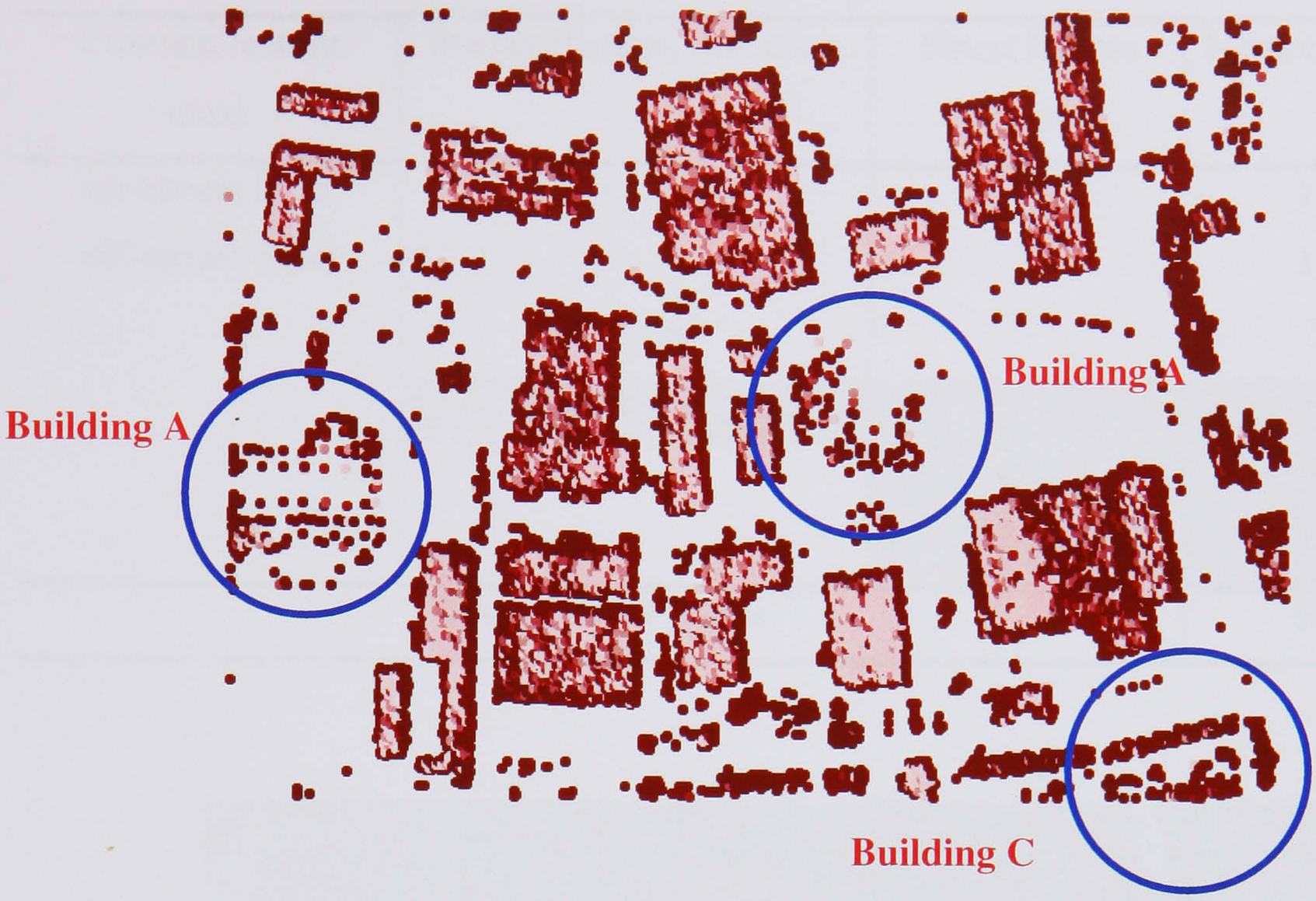


Figure 6-12. Result of the weighted normalization process of local surface curvature

Thus, a region-wise feature classification approach is suggested. The detection of trees by the local surface curvature is performed after closely connected high-rise terrain points are clustered as one region, and regional slope variance (RSV) of the clustered high-rise points is collected. A number of isolated off-terrain clusters are obtained by the TIN colouring algorithm described in §3.4.4.1. Then, an average of weighted slope variances is computed as the RSV of each high-rise cluster. Figure 6-11 (d) shows a histogram of the RSVs computed over 271 off-terrain clusters extracted by the TIN colouring algorithm. Finally, the differentiation between trees and buildings is made by simply thresholding the RSV histogram of figure 6-11 (d). If the RSV of a high-rise cluster is less than a threshold, δ_{RSV}, the cluster is classified as buildings, otherwise as trees. With δ_{RSV} of 0.8, 247 off-terrain clusters are recognized as the tree class, while 24 off-terrain clusters as the building class. Thus, the high-rise points are now re-labelled as two sub-classes of three and buildings (see table 6-6). The final classification result by the RSV is shown in figure 6-13, where detected building points are overlaid with IKONOS imagery.

Table 6-6. Result of high-rise and low-rise feature classification

Coarsest feature class	Intermediate feature class	Finest feature class	Number of points
on-terrain label	-	-	18556
off-terrain label		-	12220
	low-rise	-	3586
	high-rise		8634
		tree	1658
		building	6976
Total LIDAR points			30776

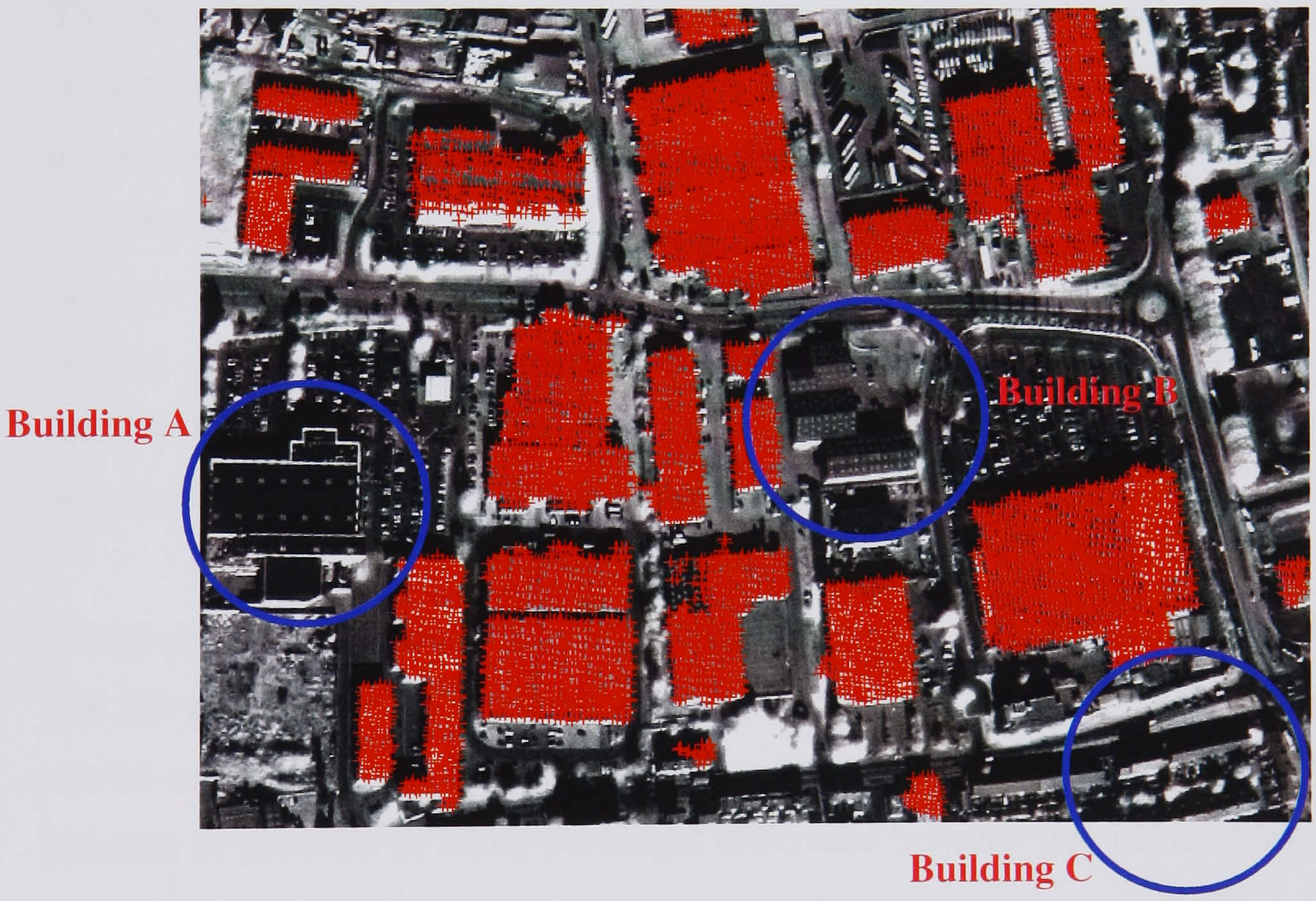


Figure 6-13. High-rise and low-rise feature classification result by RSV; red crosses represent LIDAR points labelled as high-rise features

As can be seen in figure 6-13, the RSV method successfully removes trees. However, several buildings are also excluded and thus, recognized as trees. This is mainly caused by extremely poor point density over the tops of these buildings (see figure 6-3 and blue boxes of figure 6-13). As can be seen in figure 6-3, LIDAR measurements over building-A and building-B in figure 6-13 are extremely lost due to weather condition or roof material property or careless LIDAR survey. In this case, the ratio of points located

around building edges over the total member points increases so that the RSVs become higher. As a result, the surface roughness of those buildings computed by the RSV is misinterpreted as an unstructured object such as the tree. The terraced houses (building-C) are also recognized as trees because the point density used is not high enough to cover such a narrow width of the rooftops. Similar to the case of the building-A and -B, the computed RSV is easily corrupted by the complexity of roof structure and thus, produces relatively high change rate of slope angle, which leads to the misclassification of the terraced houses. Thus, it can be concluded that the RSV classification method is sensitive to the point density and the spacing irregularity of LIDAR measurements. The success of the method requires LIDAR points evenly covers the surface of acquired objects with a high density.

The other drawback of RSV comes from the strategy of the region-wise classification. It is evident that the region-wise approach is suitable for building detection since it aims to cut off the building regions as a whole. Reciprocally, the region-wise approach might misclassify the entire buildings, only by a minor change of thresholding value. It is, however, difficult to obtain very reliable threshold for RSV since there is a range of surface roughness of the treetops and rooftops. Furthermore, the RSV may fail to classify buildings if there are many tress located near a building so that the RSV increases high enough to misclassify the building as the tree.

The problems noted above can be resolved; either if the LIDAR density is increased so that the RSV becomes less sensitive to the thresholding technique; or if last-pulse data can be acquired as well as the first-pulse so that tree points can be removed as many as possible in advance by using the height difference between two pulses. Unfortunately, the first and second solutions are not available for the current research. Therefore, other criterion to uniquely characterize tree points based on a "per-point" classification, while not being disturbed by a high complexity of rooftop structures and irregularity of point measurement, is exploited in following section.

6.4.3.2 NDVI classification

The discrimination between buildings and tress is accomplished through normalized difference vegetation index (NDVI) analysis of the IKONOS multi-spectral information. The NDVI is an established vegetation index that is routinely calculated in

the thematic mapping applications by looking at the combinatorial reflectance in the red and near-infrared bands and is defined as;

$$NDVI = \frac{NIR - R}{NIR + R} \tag{6-5}$$

where *NIR* and *R* represent data from IKONOS red and near-infrared channel respectively. Since vegetated area generally generates relatively high reflectance by near-infrared channel than by visible red channel, the NDVI values indicate the amount of green vegetation present in the pixel; higher NDVI values indicate more green vegetation (Jensen, 1986).

Figure 6-14. Result of NDVI classification

Once the NDVI map is calculated by Eq. 6-5, the high-rise features are then, divided into building and tree class using a global threshold technique of the NDVI map. This method assumes that the NDVI distribution for the high-rise feature class will have two clear bimodal distributions, one for the building and one for the tree. Figure 6-14 and figure 6-15 show the NDVI result computed from figure 6-4 and its normalized

histogram respectively. As can be seen in figure 6-15, the normalized histogram of the NDVI clear shows the presence of "vegetated" area, which peak is occurred around the maximum NDVI, i.e., ~1, and is separable from the other peak generated by the mixture of different surface materials of building rooftops.

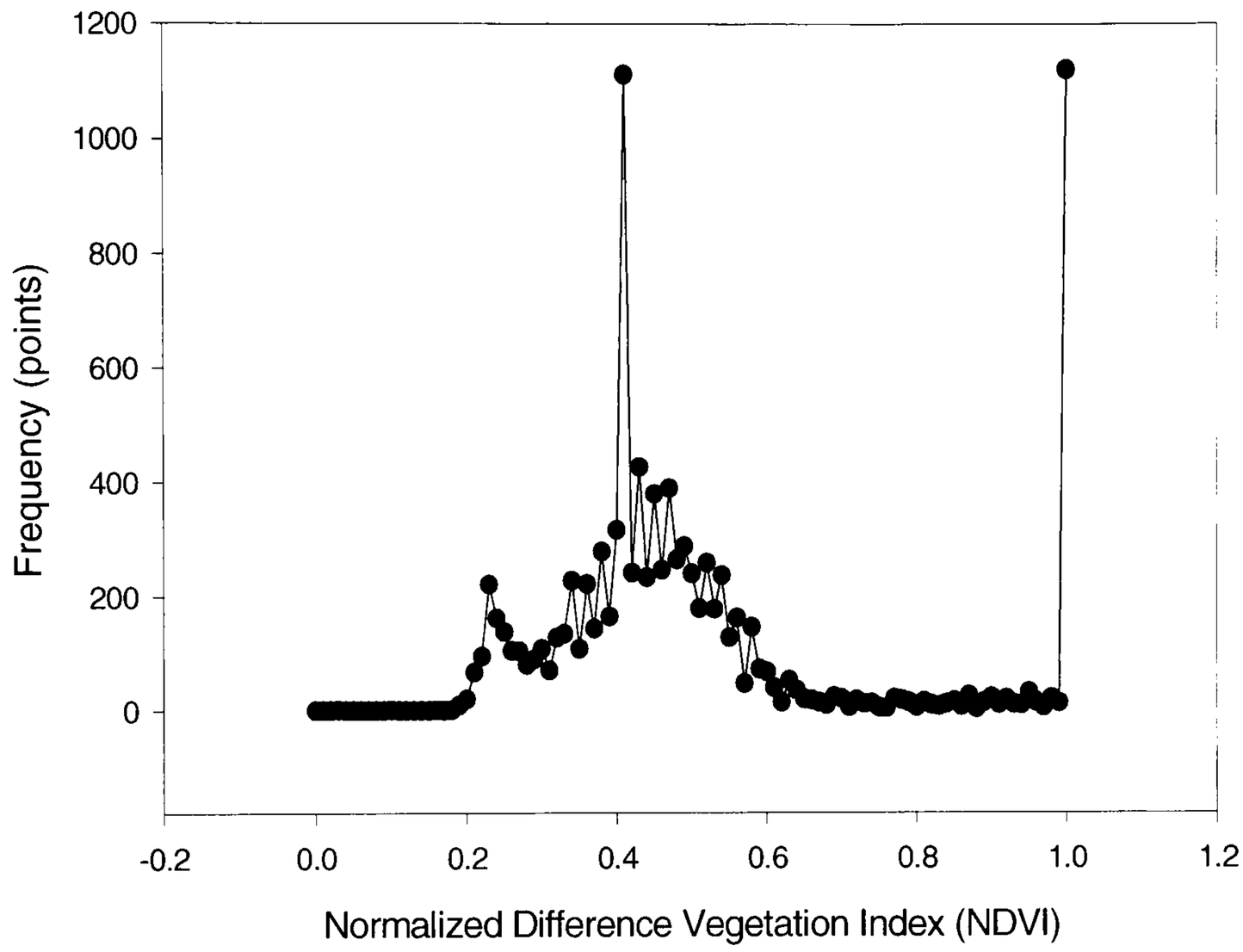


Figure 6-15. NDVI histogram over the Greenwich study area

By back-projecting the high-rise points into IKONOS image space and incorporating the NDVI information with them, the NDVI for each high-rise point is calculated from a small mask constructed around the back-projected point; the tree label is assigned to a high-rise point if any pixel of the masked points has the NDVI value larger than a pre-specified threshold, i.e., δ_{NDVI} ; otherwise, the building label is assigned. Finally, vegetated high-rise points can be removed, while building label points are retained. Figure 6-16 shows a binary image when the NDVI map is thresholded by δ_{NDVI} of 0.8. Figure 6-17 displays building label points detected when the mask size of 5x5 was used over figure 6-16.

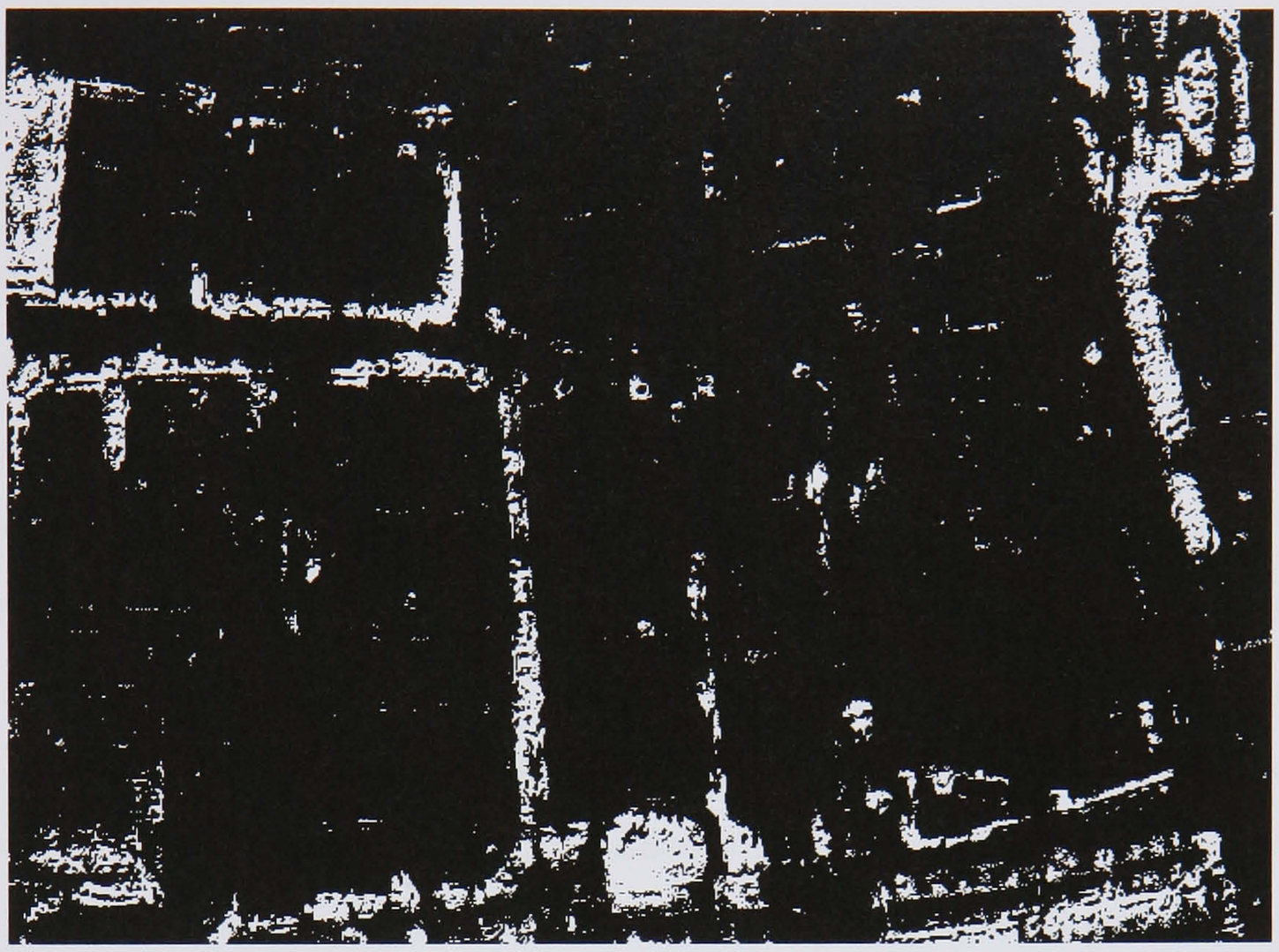


Figure 6-16. Binarized NDVI map

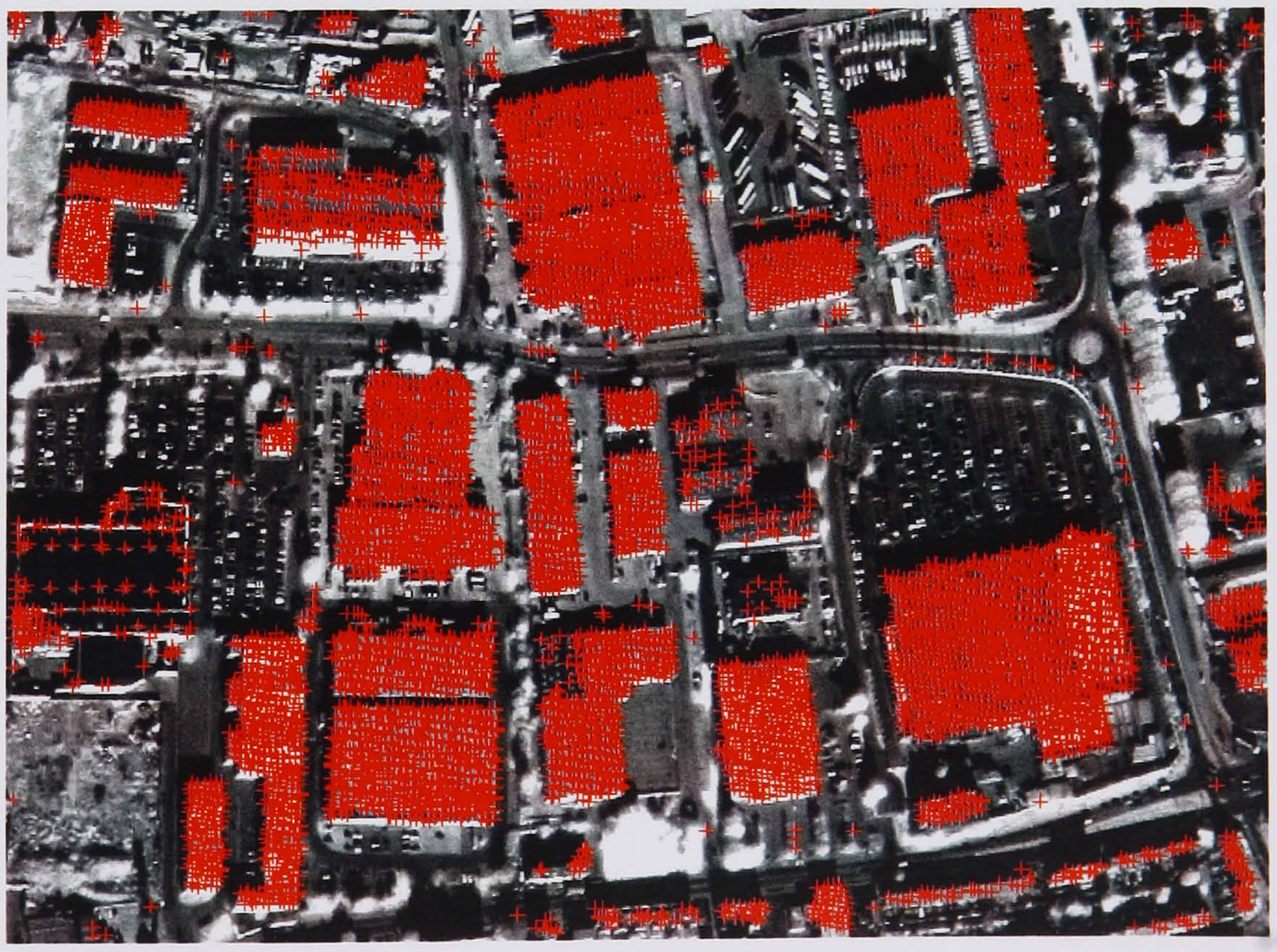


Figure 6-17. Building label points overlaid with IKONOS imagery; red dots represent LIDAR points labelled as building features

Compared to the classification method by the local surface curvature discussed in §6.4.3.1, a quick visual investigation of figure 6-17 suggested that the NDVI is a more reliable cue to differentiate vegetated features from man-made objects, which is not disturbed by the geometric structure of rooftops, but leads to the reduction of a large amount of misclassification errors. Furthermore, the computation of the NDVI is simple and the selection of δ_{NDVI} is relatively intuitive. Since the method is based upon "per-pixel" classification, the generation rate of misclassification errors is relatively insensitive to the selection of threshold value. However, the classification method relying on the NDVI threshold technique can be criticized by several factors. There are a number of cases where the NDVI of vegetated areas is very low. In such cases, trees may be misclassified as buildings. The remotely sensed reflectance of most natural surfaces is anisotropic in nature because reflectance of a surface depends on both the viewing and illumination geometry. This anisotropy is described by the BRDF (Parish., 2002). It has been widely agreed that NDVI is strongly dependent on BRDF for single images. On the other hand, the NDVI criterion is not valid when the surface material of a rooftop contains the "greenness" similar to nature vegetation.

Thus, the solution to overcome the aforementioned difficulties can be suggested; either by using multiple colour images with different viewing angles, and regularizing the sensitivity of the NDVI computation depending on the viewing and illumination geometry; or by combining a secondary cue to verify the NDVI classification result. For instance, the difference between first-pulse and last-pulse can be used as a verification cue for the NDVI classification result. That is, the tree and building points classified by the NDVI method can be verified as the analysis of "first-and-last pulse" shows significant height differences for trees, while nearly zero for building.

6.4.4 Building blob detection

Isolating the building label points and making them into individual building objects is rather straightforward. Once tree points are differentiated from building points by the NDVI method described in the previous section, those points classified into the on-terrain, low-rise, and tree objects are together assigned non-building labels. Then, building points surrounded by the non-building labels, are grouped as isolated objects (called building "blobs") by the TIN colouring algorithm mentioned in §3.4.4.1. As a result, 28 building "blobs" can be found from figure 6-18 after removing small "blobs" whose member points are less than 30 points. Further processing allows the individual building "blobs" to be bounded with rectangle polygons, and these polygons are then fed into the building extraction process, which will be discussed in the next section.

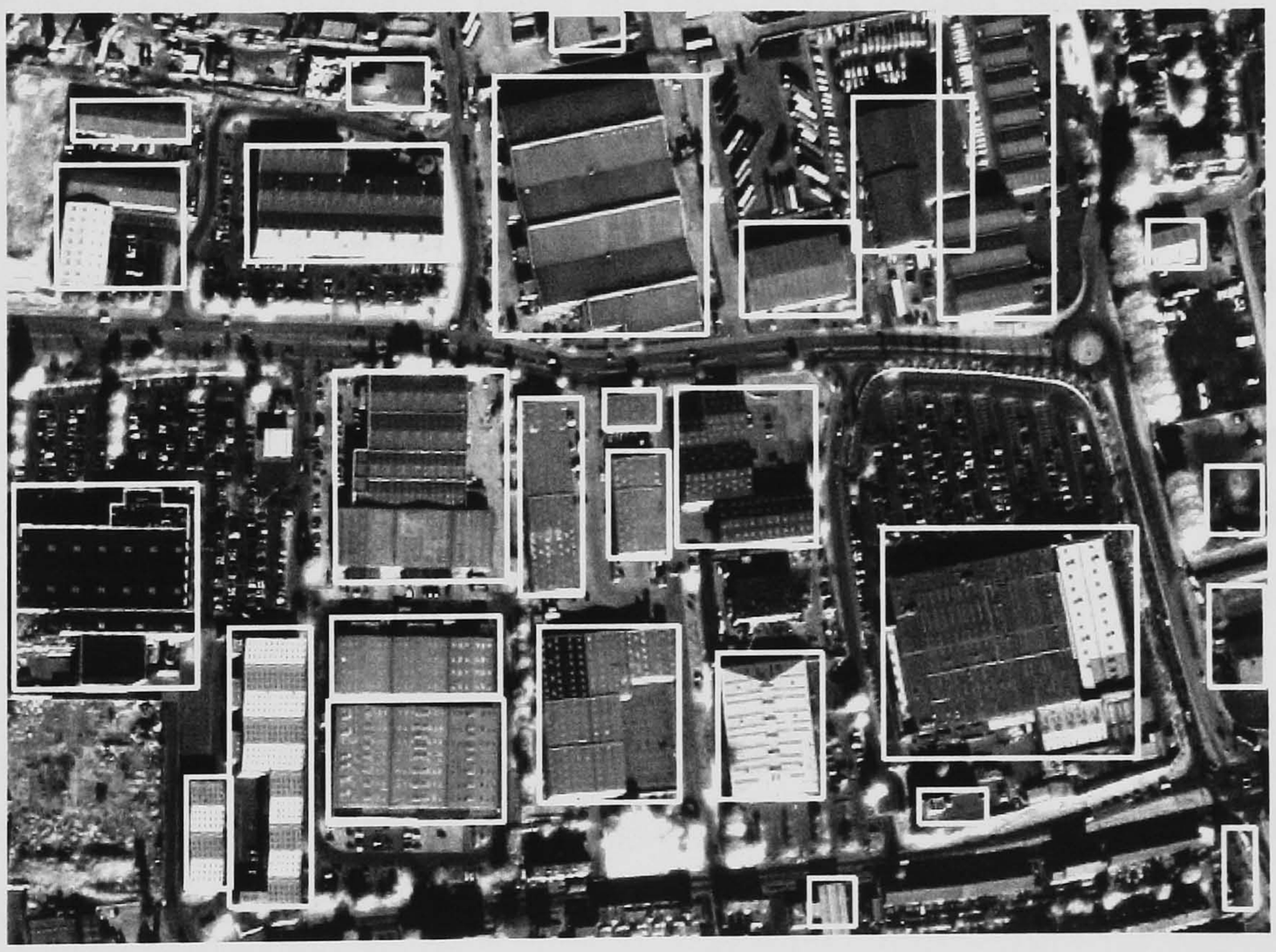


Figure 6-18. Building detection result overlaid with the Greenwich IKONOS image

6.5 Building description

This section describes a building description method, which extracts building outlines based on the result of the building detection process presented in the previous section. Generic building shape is represented as a mosaic of convex polygons. In §6.5.1, a method to extract data-driven line cues ("intensity" line cue) from IKONOS image with support of the classification result of LIDAR points is presented. In order to increase cue density, §6.5.2 describes a technique to produce model-driven line cues ("virtual" line cue) using specific building models over LIDAR space. §6.5.3 and §6.5.4 illustrate the building unit shape (BUS) organization process, which generates polygonal cues by recursive intersection of both data-driven and mode-driven linear cues and verifies them to select "building" polygons forming building shapes. As those verified "building" polygons are merged, building outlines are reconstructed. The suggested building extraction technique is evaluated using the Greenwich dataset and reference building vectors in §6.5.5.

6.5.1 "Intensity" line cue generation

It is assumed that a generic building shape consists of a set of rectilinear lines, but without the limitation of directionality. Thus, as a primary cue for building extraction, straight lines are extracted from IKONOS imagery by the Burns algorithm (Burns et al. 1986). However, since extracted line features include a number of extraneous line segments, uncorrelated to building saliencies, it is necessary to filter those distracting features so that only focused lines with significant length located around building boundaries remain. To this end, non-building (including on-terrain, low-rise, and tree) and building points labelled by the previous classification process are used to determine whether or not line primitives can be considered as boundary lines.

First, straight lines extracted by the Burns algorithm are filtered by a length criterion, by which only lines larger than pre-specified length threshold, l_d, remain for further processing. Then, two rectangle boxes with certain width, l_w, are generated along two orthogonal directions to the line vector filtered in length. The determination of boundary line can be given if non-building and building points are simultaneously found in both boxes or if only building-label points are found in one of the boxes and no LIDAR point

can be found in the other box. The latter boundary line condition is considered if a low density LIDAR dataset is used. Figure 6-19 illustrates this.

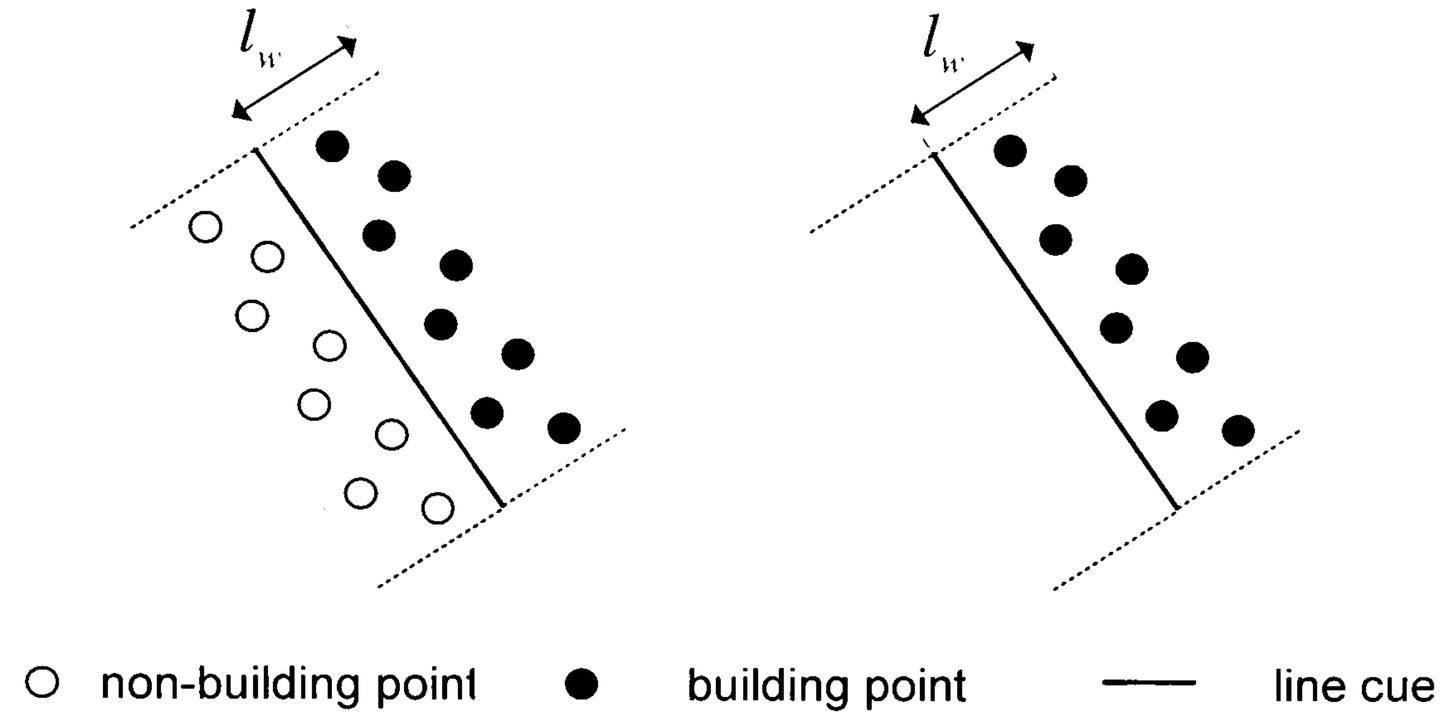
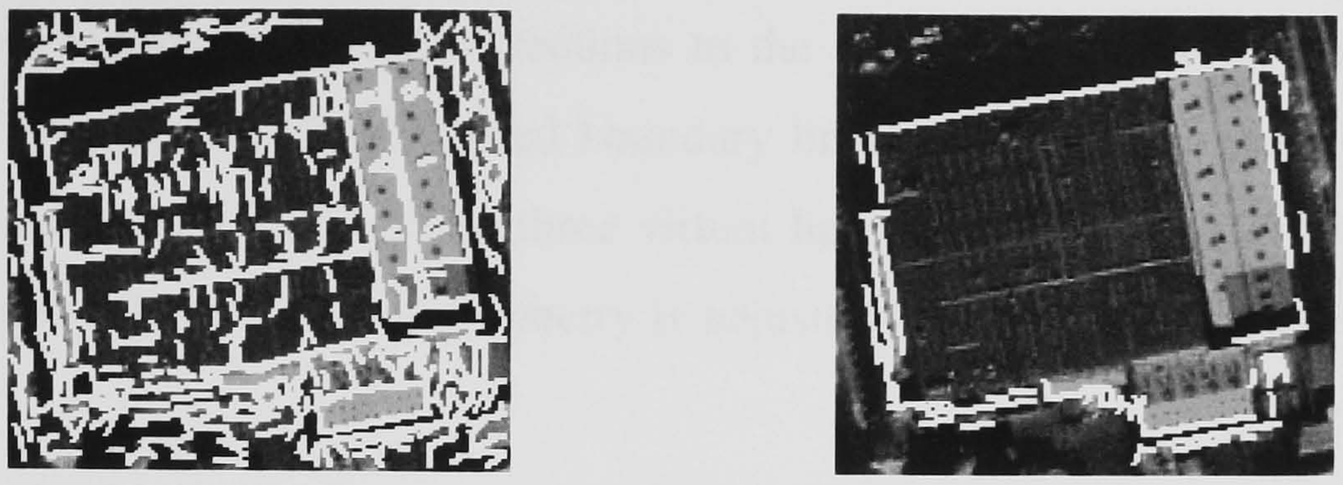


Figure 6-19. "Intensity" line verification

As a final line filtering process, a geometric disturbance corrupted by noise is regularized over boundary lines. A set of dominant line angles of boundary lines is analyzed from a gradient-weighted histogram which is quantized in 255 discrete angular units. In order to separate a weak, but significant peak from other nearby dominant angles, a hierarchical histogram-clustering method is applied. Once the dominant angle, θ_d, is obtained, lines with angle discrepancies which are less than certain angel thresholds, θ_{th}, from θ_d are found. Then, their line geometries are modified as their angles are replaced with θ_d. These modified lines do not contribute to the succeeding dominant angle analysis and the next dominant angle is obtained. In this way, a set of dominant angles is obtained, by which geometric properties of boundary lines can be regularized. Figure 6-20 shows boundary lines extracted by the support of labelled LIDAR points, after being filtered in length (l_d = 5 m) and geometrically regularized in angle threshold (θ_{th} = 30°), where a box width of l_w is chosen as 5-metres for boundary line verification. The result shows the boundary lines can be successfully extracted as the intensity line cues. Since the deficiency of LIDAR measurements on the ground nearby building boundaries in the Greenwich dataset, distracting lines located on the ground are also detected as intensity line cues. These lines will be, however, excluded during an upper level of cue generation, referred as polygon cue generation.



(a) extracted straight lines (b) filtered boundary lines

Figure 6-20. Extracted building boundary lines from IKONOS image

6.5.2 "Virtual" line cue generation

Since the current research aims to represent a generic building shape by only boundary lines, a much larger number of line cues needs to be obtained than in the case of a parametric building model. It is, however, always a bottleneck to extract such a cue density due to low contrast, shadow overcast, and occlusion effects, especially when 1-mere satellite imagery of a complex scene is solely used. Thus, new line cues are "virtually" extracted from LIDAR space in order to compensate for the lack of intensity line cue density by employing specific building models.

For extracting the "virtual" line cues, the second constraint of representing building boundaries assumes that a generic building shape is made in some degree of geometric regularity. Based upon this, for each intensity line cue, parallel lines and "U" structured lines are inferred from LIDAR space. The process aims to acquire at most three virtual lines starting from an intensity line cue, but it could fail to generate any virtual line cue. The main idea of virtual line detection is that a small virtual box is generated from each intensity line and it grows over building roof so that building points are maximally captured without including any non-building point.

First, a box growing direction, pointing to the location of parallel boundary line is determined. To this end, a small virtual box is generated with a width of l_w from the selected intensity line in the same way of detecting boundary lines in §6.5.1. To that direction, the virtual box grows until it comes across any on-terrain point (see figure 6-21 (a)). Then, it de-grows in order to have maximum building points while in its

minimum size (see figure 6-21 (b)). In this way, the virtual box is expanded, but at this time, towards to two orthogonal directions to the parallel boundary line detected (see figure 6-21 (c)). Thus, "U" structured boundary lines made with the parallel boundary line can be detected. Finally, these three virtual lines detected are back-projected onto image space and then, their line geometry is adjusted by gradient weighted least-square method.

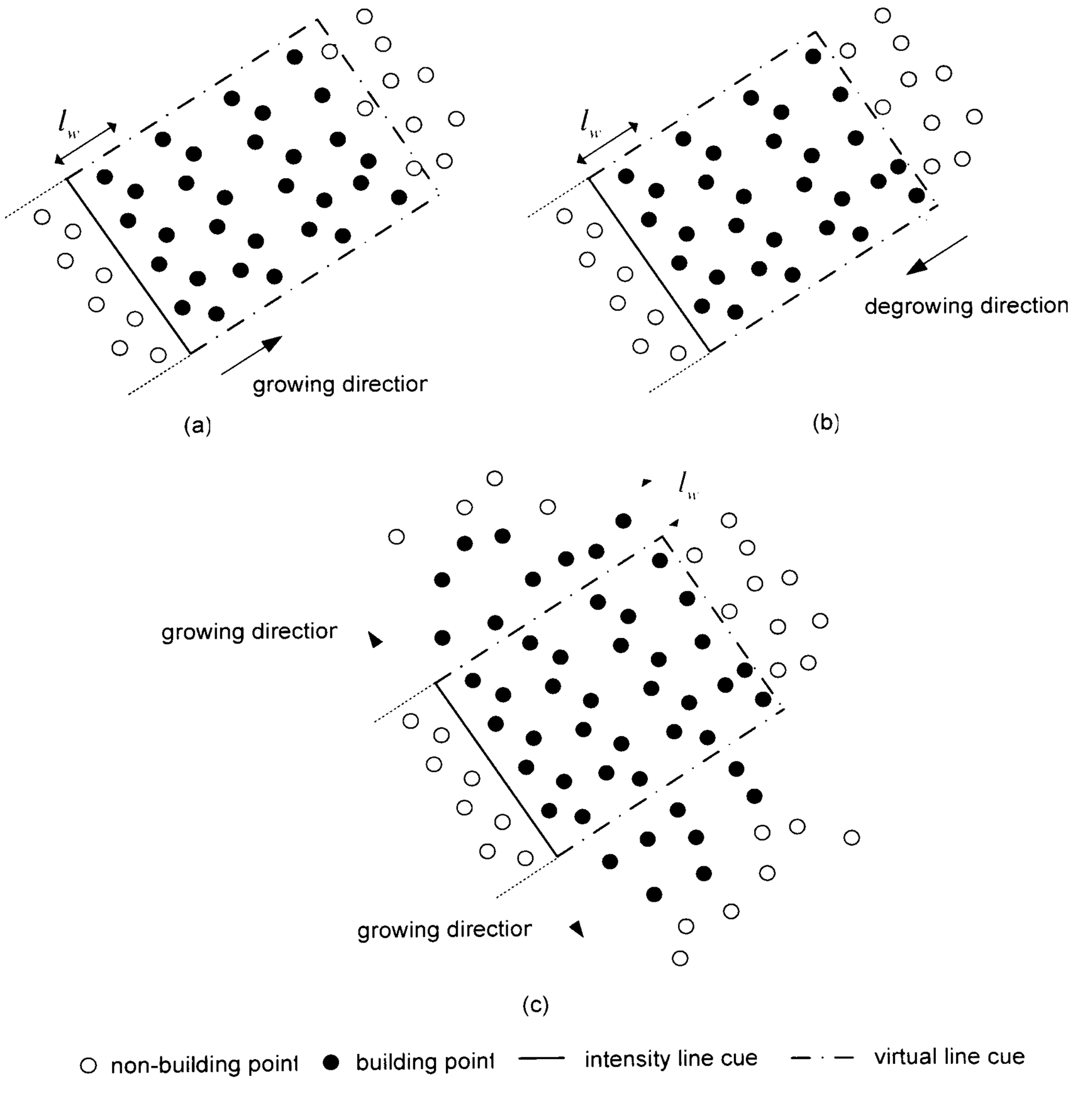
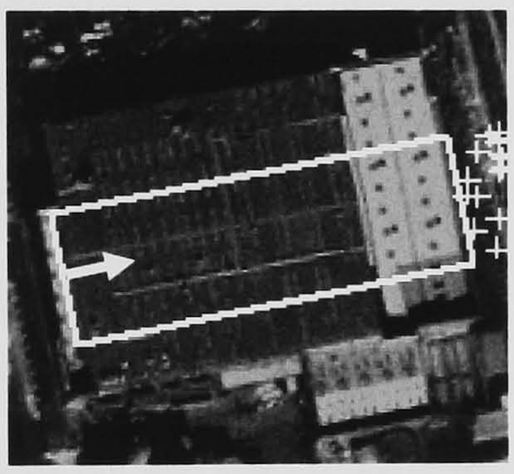


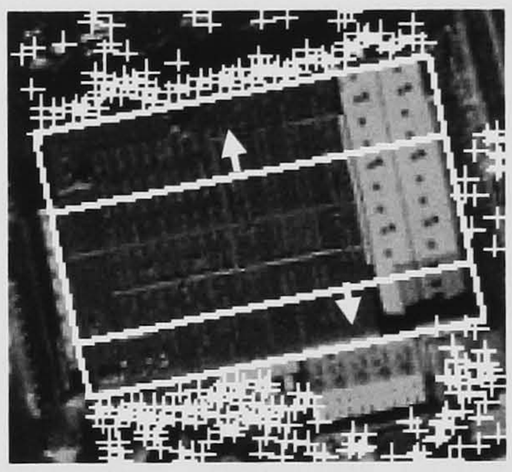
Figure 6-21. Illustration of virtual line cue generation

When l_w is chosen as 5-metres, a procedure of detecting three virtual line cues from one intensity line can be illustrated in figure 6-22 (a)-(c). As remaining intensity lines cues are involved, figure 6-22 (d) shows the entire virtual line cues detected. It is noteworthy that the assumption of geometric regularity for a building shape constraint will be used as a weak constraint in the suggested building extraction system since the geometry of a

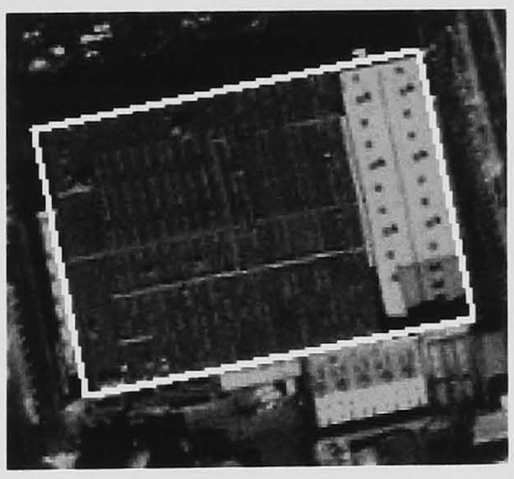
generic building shape may not have any symmetric property or erroneous lines may be included as intensity lines. For this reason, only a portion of virtual cues could be involved to recover significant boundary segments missed in intensity line cue generation. It is, however, subject to the degree of complexity of individual buildings and a false alarm rate of boundary line detection as to what percentage of virtual lines cues can be used. Therefore, a verification process of virtual line cues is necessary. In the following section, it will be automatically determined at polygonal cue generation, whether or not a virtual line cue should be used in boundary representation of polyhedral building shape.



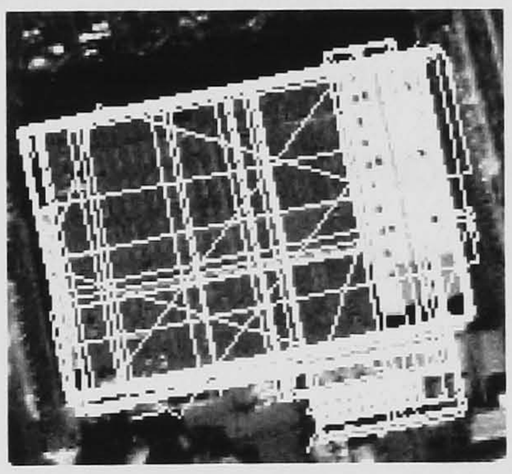
(a) parallel line detection



(b) "U" structure line detection



(c) gradient fitted virtual lines



(d) entire virtual lines detected

Figure 6-22. Results of virtual line cue generation; white lines represents detected virtual lines, white arrow points to the virtual box growing direction and white cross represents non-building points.

6.5.3 BUS generation

Once intensity line and virtual line are extracted, a set of convex polygon cues is generated by a recursive intersection of both lines. Only the polygons comprising "significant" parts of building shape are verified as "building" polygon. The boundary representation of generic building shape is reconstructed by a collection of "building" polygons. A line cue grouping process to generate polygon cues is implemented by the BSP method, which was described in §5.4.3. Considering the contribution of LIDAR data, the method of BUS space generation by the BSP is modified with a different strategy as follows:

- An initial polygon for the BSP is automatically generated by the building detection process, and is delineated with a rectangle;
- Hyperlines to partition given polygon and to generate two child convex polygons are constructed by "intensity" line cues and "virtual" line cues;
- Instead of excluding a hyperline after it is finally used for partitioning a polygon, all the hyperlines are recursively used for obtaining the "best" partitioning result until the process is terminated; and,
- A hyperline which leads to the "best" partitioning result of a polygon is determined as an optimum solution of BSP by a partition scoring function. This function considers a level-of-detail (LoD) strategy so that the most "significant" part of building structure is generated as a partitioning result first, and a less "significant" part is obtained later.

Figure 6-23 illustrates the overall partitioning scheme to generate BUS space. Suppose that we have an initial polygon with rectangle geometry, P^0, wherein LIDAR points are distributed with building and non-building label. The P^0 polygon is generated by the building detection process and an individual building object is surrounded by this polygon. Once P^0 is generated, all vertices comprising P^0 are stored as a root node of BSP tree for further recursive partitioning (see figure 6-23 (a)).

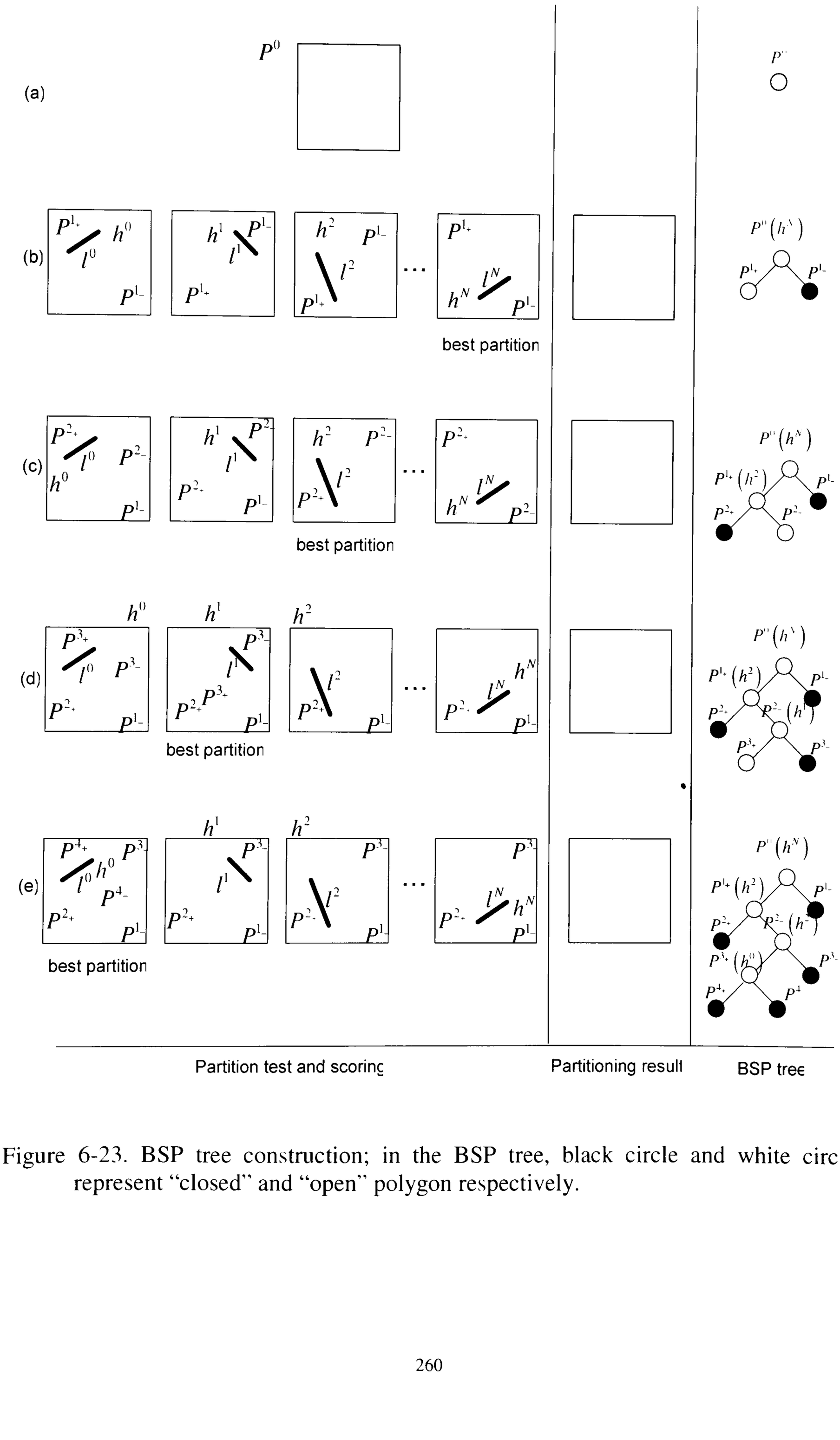


Figure 6-23. BSP tree construction; in the BSP tree, black circle and white circle represent "closed" and "open" polygon respectively.

For recursive partitioning, a set of hyperlines, $\{h^i : i = 1,..., N\}$, are computed as P^0 is intersected respectively by a set of line segments, $\{l^i : i = 1,..., N\}$, which is constructed by integrating the intensity and virtual line cues described in §6.5.1 and §6.5.2. After setting up the hyperline list, a process to partition P^0 with hyperlines starts. This partitioning process consists of two procedures; *polygon classification* and *partition scoring*.

Polygon classification is a process to determine whether or not the partitioning process is triggered over a given polygon, P^i. A polygon, P^i, is classified into a number of polygon classes; "empty", "open", "closed", "pseudo-closed", "garbage" polygon. These polygon classes are pre-determined depending on the labelling attributes of the member points of P^i or point density of the member points of P^i or geometric property of P^i as follows:

- ***"Empty" polygon:*** P^i is classified as "empty" polygon if there is no member point within P^i (see figure 6-24 (a)).
- ***"Open" polygon***: P^i is classified as "open" polygon if the member points of P^i are attributed to both building and non-building labels (see figure 6-24 (b)).
- ***"Closed" polygon***: P^i is classified as "closed" polygon if the member points of P^i are attributed to only building label (see figure 6-24 (c)).
- ***"Pseudo-closed" polygon***: P^i is classified as "pseudo-closed" polygon if the member points of P^i are attributed to only building label, and the point density of P^i, $d_{pt}\left(P^i\right)$, is less than d_{th} (see figure 6-24 (d)), where $d_{pt}\left(P^i\right)$ is determined by

$$d_{pt} = \frac{N_{mem}\left(P^i\right)}{A^i} \tag{6-6}$$

where $N_{mem}\left(P^i\right)$ and A^i are the number of member points and the area of P^i respectively. Given n vertices of $P^i\left(x_i, y_i\right)$, $i = 0, ..., n$, with $x_0 = x_n$ and $y_0 = y_n$, the following formula can be used for calculating A^i:

$$A^i = \frac{1}{2}\left(\begin{vmatrix} x_1 & x_2 \\ y_1 & y_2 \end{vmatrix} + \begin{vmatrix} x_1 & x_2 \\ y_1 & y_2 \end{vmatrix} + \ldots + \begin{vmatrix} x_1 & x_2 \\ y_1 & y_2 \end{vmatrix}\right) \tag{6-7}$$

where $\left|A^i\right|$ denotes a determinant. This can be written by

$$A^i = \frac{1}{2}\left(x_1 y_2 - x_2 y_1 + x_2 y_3 - x_3 y_2 + \ldots + x_{n-1} y_n - x_n y_{n-1} + x_n y_1 - x_1 y_n\right) \tag{6-8}$$

where A^i is defined to be positive if the points are arranged in a counter-clockwise order, and negative if they are in clockwise order.

- ***"Garbage" polygon***: P^i is classified as "garbage" polygon if the member points of P^i are attributed to both building and non-building labels, and any lateral length or the area P^i is less than a certain threshold, i.e., l_{th} and a_{th} respectively (see figure 6-24 (e)).

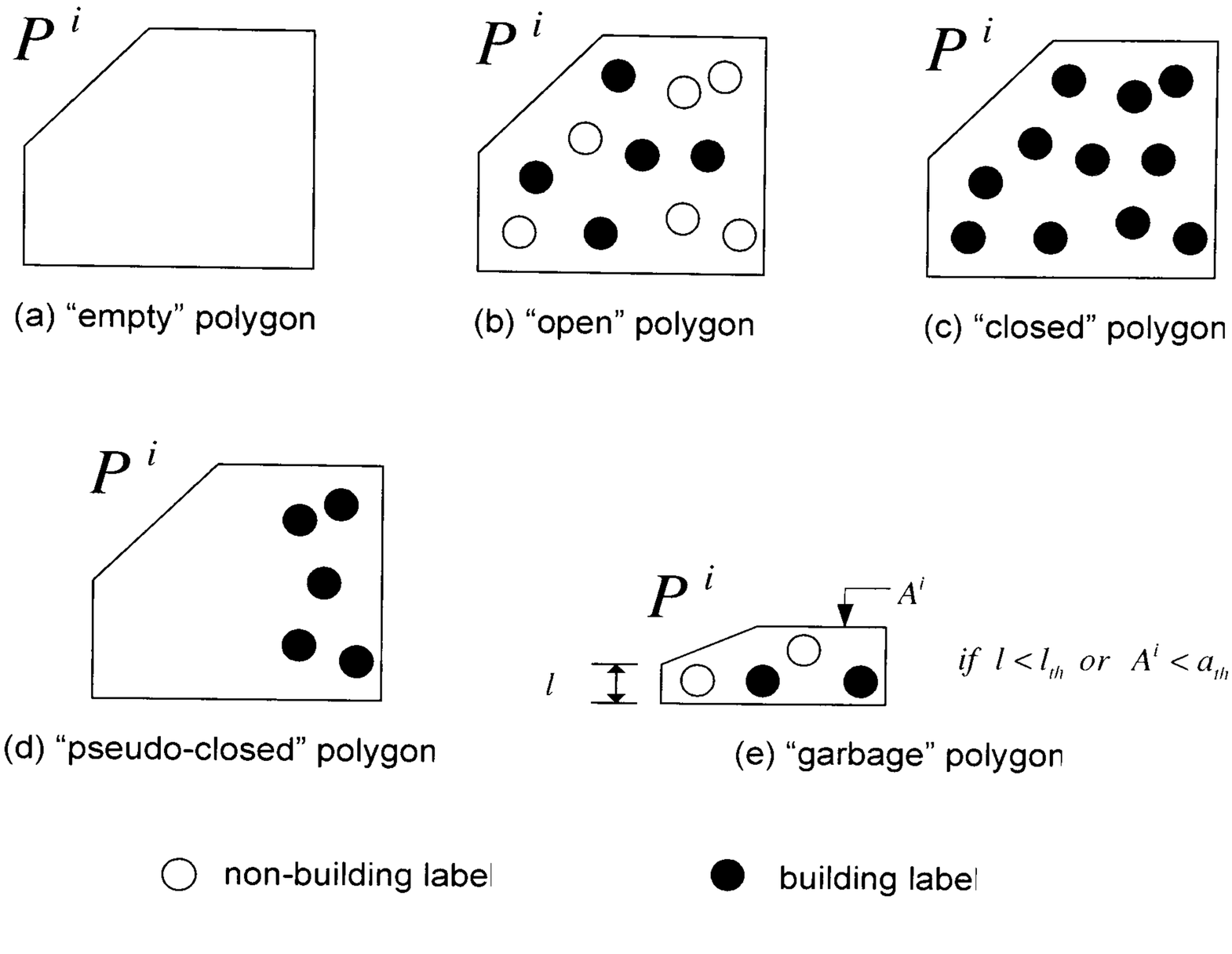


Figure 6-24. Illustration of polygon classification

A triggering and terminating condition of the partitioning process over P^i is determined depending on the polygon classification result. Basically, P^i is partitioned further if it is classified as “open” polygon, whereas the partitioning process over P^i is rejected if P^i is classified as one of “closed” polygon or “empty” polygon.

In addition, there are some special conditions for partitioning P^i. Due to weather conditions or surface material property or careless LIDAR survey, LIDAR points might be acquired with coarse point density so that no non-building point is detected within P^i. In this case, P^i is classified as the “closed” polygon and it is “earlier” terminated, though it needs to be partitioned more. To prevent this situation, the “pseudo-closed” polygon, which is the special case of the “closed” polygon with coarse point density, is also considered as a triggering condition to partition P^i. In addition, it is necessary to prevent the partitioning method from generating an ignorable building part. To this end, P^i is rejected to partition P^i if it is classified as the “garbage” polygon.

Once the partitioning of a given P^i is determined through the polygon classification. The second step is *partition scoring*. This process determines a hyperline, h^*, to generate the "best" partitioning result of P^i from the hyperline list. The selection of h^* is achieved by a partition scoring function. That is, all the hyperlines are tested to obtain the "best" partition of P^i and the partitioning result generated by each hyperline is evaluated by a partition scoring function. A hyperline, h^*, with the highest partitioning score is finally selected to partition P^i.

Figure 6-23 (b) shows a procedure to obtain the "best" result of the initial polygon, P^0. A hyperline candidate, h^i, is selected from the hyperline list, by which P^0 is divided into the positive and negative planes, P^{l+} and P^{l-}, reflecting whether the dot product of each vertex point of P^0 with h^i is negative or positive. Then, a partition scoring function, H, evaluates the partitioning results respectively over P^{l+} and P^{l-} generated by h^i, and then determines a final score for h^i as the maximum value of the partitioning results over P^{l+} and P^{l-}. This scoring function, H, can be described as follows:

$$H\left(P^0; h^i, l^i\right) = \arg\max\left(H\left(P^{l+}; h^i, l^i\right), H\left(P^{l-}; h^i, l^i\right)\right) \tag{6-9}$$

In Eq. 6-9, H assigns a maximum score to h^i if it produces the best partitioning result, whereas a minimum score for the worst partitioning result. Also, H differently computes scores over P^{l+} and P^{l-} depending the polygon class of P^0. If P^0 is classified as the "open" polygon, H computes partitioning scores according to a bias degree of label distribution over P^{l+} and P^{l-} divided by h^i; if a more biased label distribution is generated, higher score is given. In other words, H for "open" polygon computes higher partitioning score when a "closed" polygon with larger area is produced by h^i (see figure 6-24 (a)). The partition scoring function, H, for "open" polygon can be described by

$$H\left(P^{1_+};h^i,l^i\right)=\frac{1}{2}\left\{\frac{N_{bld}\left(P^{1_+};h^i,l^i\right)}{N_{bld}\left(P^0;h^i,l^i\right)}+\frac{N_{non-bld}\left(P^{1_-};h^i,l^i\right)}{N_{non-bld}\left(P^0;h^i,l^i\right)}\right\}$$
$$H\left(P^{1_-};h^i,l^i\right)=\frac{1}{2}\left\{\frac{N_{bld}\left(P^{1_-};h^i,l^i\right)}{N_{bld}\left(P^0;h^i,l^i\right)}+\frac{N_{non-bld}\left(P^{1_+};h^i,l^i\right)}{N_{non-bld}\left(P^0;h^i,l^i\right)}\right\} \quad (6\text{-}10)$$

where $N_{non\text{-}bld}$ and N_{bld} are functions to count numbers of building labels and non-building labels belonging to a corresponding polygon. If P^0 is classified as the "pseudo-closed" polygon, H computes the partitioning score by an area ratio of child "empty" polygon over P^0 when either of P^{i+} and P^{i-} is recognized as an "empty" polygon, otherwise the null value is assigned as the partitioning score for P^0 by H. The "best" partition of P^i can be obtained when an "empty" polygon with the largest area is produced by h^i (see figure 6-24 (b)). The partition scoring function, H, for "pseudo-closed" polygon can be described by

$$\begin{cases} H\left(P^{1_+};h^i,l^i\right)=\dfrac{A\left(P^{1_+}\right)}{A\left(P^o\right)} & if\ \ P^{1_+}=\text{"}empty\text{"}\ polygon \\ H\left(P^{1_-};h^i,l^i\right)=\dfrac{A\left(P^{1_-}\right)}{A\left(P^o\right)} & if\ \ P^{1_-}=\text{"}empty\text{"}\ polygon \end{cases}$$

$$\begin{cases} H\left(P^{1_+};h^i,l^i\right)=0 & if\ \ P^{1_+}\neq\text{"}empty\text{"}\ polygon \\ H\left(P^{1_-};h^i,l^i\right)=0 & if\ \ P^{1_-}\neq\text{"}empty\text{"}\ polygon \end{cases} \quad (6\text{-}11)$$

where $A()$ is a functions to calculate the area of corresponding polygon by Eq 6-8.

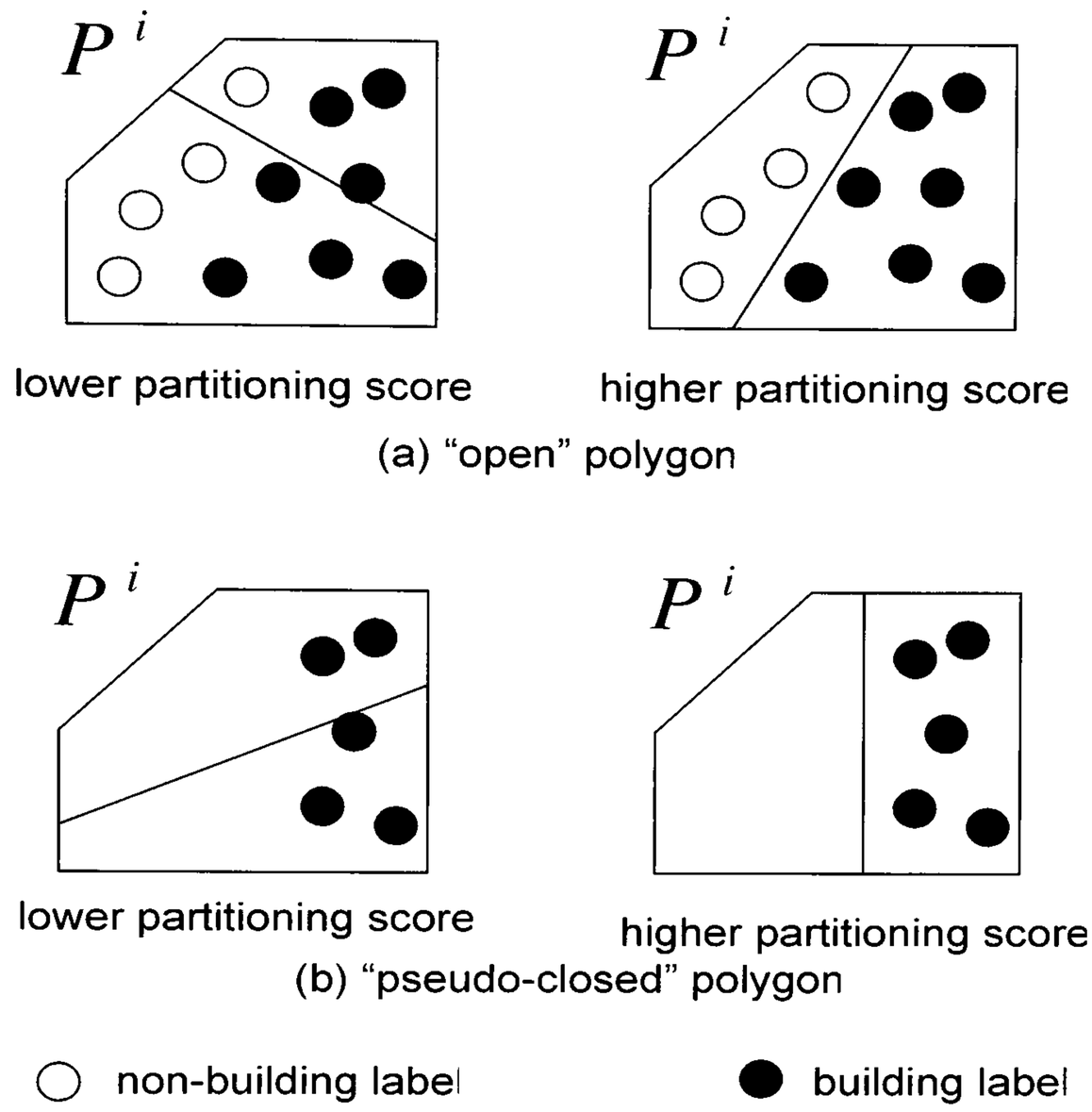


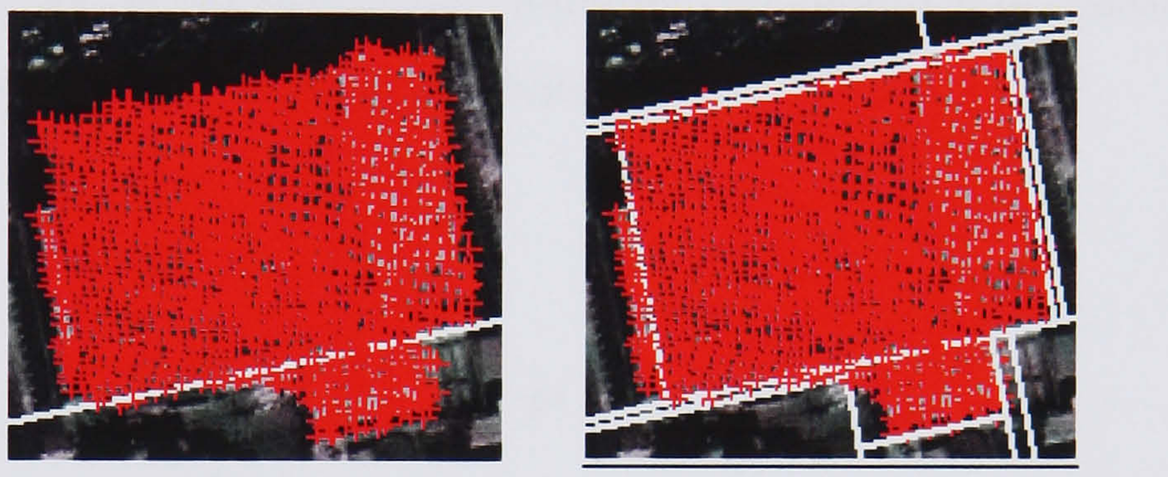
Figure 6-25. Illustration of partition scoring; P^i is a parent convex polygon to be partitioned and h^i is a hyperline to partition P^i

Once the partitioning scores for h^i are computed by Eq. 6-10 or Eq. 6-11, remaining hyperlines are sequentially selected from the hyperline list and their partitioning scores are measured by H. As can be seen in figure 6-23 (b), a hyperline, h^N, with the maximum partitioning score is finally selected to partition P^0. Then, geometric information of P^0 and h^N are stored as a root node of BSP tree, which is expanded as new child nodes with vertices of P^{1+} and P^{1-} are added to the root node for further recursive partitioning. The same method used for the partition of P^0 is applied to P^{1+} and P^{1-} respectively, but to only an "open" or "pseudo-closed" polygon. In figure 6-23 (b), P^{1+} is classified as the "closed" polygon and P^{1-} is classified as one of the "open" polygon and "pseudo-closed" polygon. Thus, the partitioning process over P^{1-} is triggered, and the partition of P^{1-} by a hyperline, h^2, is selected as the "best" result. Similar to the case that the BSP is expended by h^N, the BSP tree of figure 6-23 (b) is expanded with new child polygons, P^{2+} and P^{2-}, and the selected hyperline, h^2 (see

figure 6-23 (c)). This process continues until no leaf node of the BSP tree can be partitioned by hyperlines (see figure 6-23 (d) and (e)).

6.5.4 BUS grouping

Figure 6-26 (a) – (c) shows an example how the BUS space with a set of convex polygons is generated by the recursive partition of an initial polygon as described in the previous section.



(a) initial partitioning (b) intermediate partitioning

(c) final BSP partitioning (d) "open" polygon filtering

Figure 6-26. Illustration of polygon cue generation and grouping (white line: selected hyperlines, red cross: building-label points)

Once the BUS space is generated by expanding a BSP tree, final leaves of the BSP tree are collected. A heuristic filtering is applied to them so that only "building" polygons remain. A convex polygon of final leaves of the BSP tree is verified as the "building" polygon by following rules:

- A polygon, P^i, is verified as the "building" polygon If it is classified as "closed" polygon, satisfying following conditions:

$$N_{mem}\left(P^i\right) > n_{th} \;\; and \;\; d_{pt}\left(P^i\right) > \gamma \times d_{th};\;\; P^i = "closed" \; polygon \qquad (6\text{-}12)$$

where N_{mem} is the number of member points of P^i; n_{th} is a member point threshold; d_{pt} is the point density of P^i, which can be computed by Eq. 6-6; γ is a control parameter ($0 \le \gamma \le 1$); d_{th} is a point density threshold.

- A polygon, P^i, is verified as the "building" polygon if it is classified as "open" polygon, satisfying following conditions:

$$\rho_{pt}\left(P^i\right) = \frac{N_{bld}\left(P^i\right)}{N_{mem}\left(P^i\right)} > \rho_{th};\;\; P^i = "open" \; polygon \qquad (6\text{-}13)$$

where ρ_{pt} is a point ratio of building labels over the total number of member points of P^i; N_{bld} and N_{mem} are functions to count numbers of building labels and non-building labels belonging to P^i.

In the aforementioned heuristic filtering, a "closed" polygon, which has a very small number of member points with coarse point density, is ignored as part of building object, while an "open" polygon, which has building labels with a significant ratio, it is classified as the "building" polygon, though it includes non-building labels. Once all "building" polygons are found, they are merged together and building boundaries are reconstructed. A height of the building roof can be also computed as the average or median, or maximum or minimum of the heights of LIDAR points, which are encompassed by the reconstructed building boundaries. Thus, the prismatic building model can be reconstructed. Figure 6-26 (d) shows a final result of the BUS grouping process.

6.6 Building extraction results

The aim of this section is to assess the positional accuracy of the building boundaries produced by the proposed building extraction technique and evaluate their cartographic completeness. In §6.6.1, the building extraction results over the Greenwich dataset are presented with a comprehensive description of the parameters selected for this technique. The performance of the proposed building extraction technique is evaluated by comparing the extracted building boundaries to the Ordnance Survey MasterMap® in §6.6.2.

6.6.1 Selection of parameters and building extraction results

The BUS organization method presented in the previous section is applied to the building detection result of figure 6-18, in which 28 buildings with building labels more than 30 points were automatically detected and bounded with rectangles. Some parameters for building extraction by the BUS organization method were pre-determined, which are summarized in table 6-7.

Table 6-7. BUS organization parameter for the Greenwich data

Purpose	Parameter	Value	Determination
"intensity" line extraction	l_d	5 (meter)	Pre-defined
	θ_{th}	30 (degree)	Pre-defined
	l_w	5 (meter)	Pre-defined
"virtual" line extraction	l_w	5 (meter)	Pre-defined
BUS space generation	l_{th}	5 (meter)	Pre-defined
	a_{th}	50 (m^2)	Pre-defined
	d_{th}	0.1 (points/ m^2)	Automatically
BUS grouping	n_{th}	5 (points)	Pre-defined
	d_{th}	0.1 (points/ m^2)	Automatically
	γ	0.6	Pre-defined
	ρ_{th}	0.6	Pre-defined

For extracting the "intensity" lines, straight lines, whose length is larger than l_d = 5 (m) are extracted from the Greenwich IKONOS image. Two rectangle boxes with the width, l_w = 5 (m), are created from each straight line and the building edge lines are found by the labelling attributes of the LIDAR points within the boxes. The dominant angles of those edge lines are analyzed with the angular resolution of θ_{th} = 30°, by which the geometries of the edge lines are regularized. Then, two rectangle boxes with the width, l_w = 5 (m), are again created from each edge line in order to determine a box growing direction towards building rooftop, and parallel lines and orthogonal lines for each edge line are created. Here, l_d is chosen considering the spatial resolution of IKONOS image and a detailed level of description of building shape required; l_w is selected depending on the point density and irregularity of LIDAR measurements; θ_{th} is determined considering a degree of complexity of building shape to cope with. In particular, θ_{th} is subject to the point density of LIDAR measurements used. Smaller angular resolution for θ_{th} can be chosen over a LIDAR data with higher point density, which can cope with higher complexity of building shape.

For the generation of BUS space, the "pseudo-closed" polygon is pre-defined with the point density threshold, d_{th}, which is given as an average point density of LIDAR dataset used. In this study, the d_{th} for the Greenwich LIDAR DSM is computed as 0.1 (points/m^2). The definition of the "garbage" polygon is given by l_{th} of 3-metres and a_{th} of 50 square metres. If a LIDAR data with higher point density (i.e., higher d_{th}) is used, smaller l_{th} and a_{th} can be selected and thus, higher level of details of building shape can be delineated. For the BUS grouping process, an "open" polygon is verified as a building part if building-label points are included over a given polygon with the point ratio of ρ_{th} larger than 0.6. In order to remove erroneous "closed" polygons, γ and n_{th} are selected as 0.6 and 5 respectively. A level of details to delineate a building shape can be controlled by modifying the parameters used for the BUS grouping process. Since the point spacing irregularity of the Greenwich LIDAR DSM is extremely high, the roles of the parameter for the BUS space generating and grouping process are important in the current implementation. However, the selection of those parameters can be more stabilized if evenly distributed LIDAR with higher point density is used.

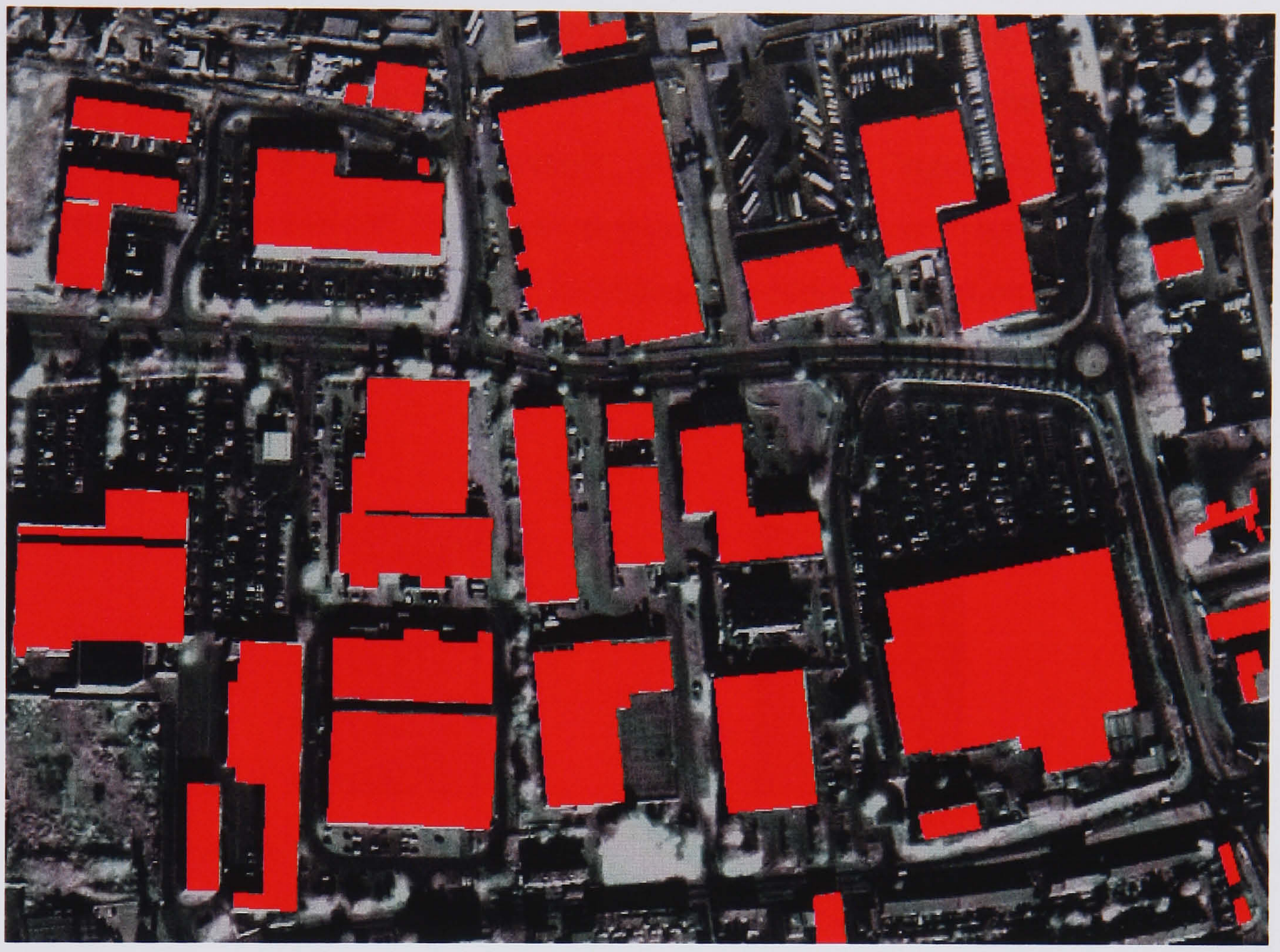


Figure 6-27. UCL building map over the Greenwich area

Figure 6-27 shows a building extraction result over the Greenwich dataset (referred as UCL building map), which was generated by the BUS organization method with the parameters of table 6-7. As can be seen in this figure, the boundaries of most building shapes are successfully reconstructed from initial building "blobs" generated by the building detection process. In the Greenwich scene, the contrast between the ground and buildings is not so high, and most building rooftops show complex texture and different shapes comprising a number of planar surfaces. In these circumstances, a large amount of erroneous linear cues, which are not relevant to building boundaries, could be generated from each building "blob". However, the suggested method efficiently prunes those distracting linear cues, and retains building edge lines. This suggests that a "good" balance between data-driven and model driven cues for building extraction can be achieved by BUS generation process. In particular, it can be concluded that the contribution of "virtual" line cues is obvious for building extraction. For several building objects, we can observe that extended building structures with small extent are reconstructed even though those parts are occluded or have very low contrast (see figure

6-28). It is difficult to obtain sufficient linear cues for describing such a small object, relying on a data-driven process only. This difficulty can be overcome by employing a simple building model with minimal use of cue grouping rules. Furthermore, the recursive part segmentation of initial convex polygon for extracting building boundaries adopts a level-of-detail (LoD) strategy, which extracts "significant" building part comprising larger size of "building" polygon first and less "significant" part with smaller extent later. This is suitable for reducing building extraction errors, which are restricted only to minor building structures.

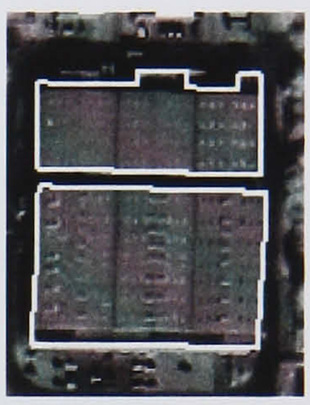



Figure 6-28. Building extraction results of small extended building structures with low contrast.

6.6.2 Quality assessment

The positional accuracy of a digital representation of a building object can be defined through measuring the difference between the apparent locations of the extracted building outlines and their true location. The task of this section is to assess the positional accuracy of the building boundaries produced by the proposed building extraction technique, and evaluate their cartographic completeness in relation to the reference building polygons provided by the Ordnance Survey.

The Ordnance Survey provided reference building polygons for this study. These data were provided in MasterMap® format. MasterMap® is Ordnance Survey's new large-scale mapping product, which encompasses a new referencing framework to deliver data more efficiently, in an attempt to provide the definitive digital map database of Great Britain. MasterMap® is the end-product of a comprehensive recompilation of Ordnance Survey's unstructured, line-based Land-Line® product, into a topologically structured format (Ordnance Survey, 2003). Although MasterMap® has no definitive scale, the initial feature data has been engineered from the Land-Line® product, and the accuracy and resolution of the geometry derive from that large-scale product. Therefore,

the original scale of the topological features correspond to that of the 1:1,250 Land-line® product, and the absolute accuracy of the MasterMap® product is currently ± 0.5 metres.

MasterMap® is ordinarily provided in GML2 (Geography Mark-up Language v2.0) format, which is an OpenGIS Consortium (OGC) standard for data distribution. For this study, the MasterMap® data was converted from GML2 to ARC Shape file format, and was re-projected from the OSGB36 to WGS84 co-ordinate system to allow an accurate overlay of the UCL building map and OS data. The MasterMap® building polygon data can be seen in figure 6-30 overlaid on one band of the IKONOS image.

Figure 6-29. OS MasterMap® building polygons

Although the detail and accuracy of the OS MasterMap® is ideal for this study, there is an inconvenient characteristic of the OS MasterMap® that presents difficulties in the accuracy assessments. The problem with the OS truth data is that it contains some significant differences in comparison to the UCL building map. This is because the OS

data was constructed at a different time to the acquisition of the IKONOS image and LIDAR data, from which the UCL data was generated. Compared to the UCL building map of figure 6-27, this temporal difference is clearly illustrated with large buildings (Building-A, Building-B, and Builidng-C in figure 6-29) that are not present in the OS data, but are clearly present in the IKONOS image and LIDAR data and therefore present in the UCL building map. This can be seen in figure 6-30. Since this research has an interest to evaluate the positional accuracy of the UCL building map, rather than the OS MasterMap®, those inherent faults in the OS data were removed from the UCL building maps (see figure 6-31). As outlined earlier, the scale of the features in the OS MasterMap® has been compiled at a larger scale than the one of IKONOS image. As a result, very small features cannot be clearly recognized in the IKONOS image. Thus, those small features, whose member points are less than 100 points, were removed from the OS data (see figure 6-32).

Figure 6-30. Inherent faults in OS MasterMap®; each row shows a building which is not present in the OS MasterMap®, but can be clearly seen in the IKONOS image (first column), LIDAR DSM (second column), and UCL building map (third column)

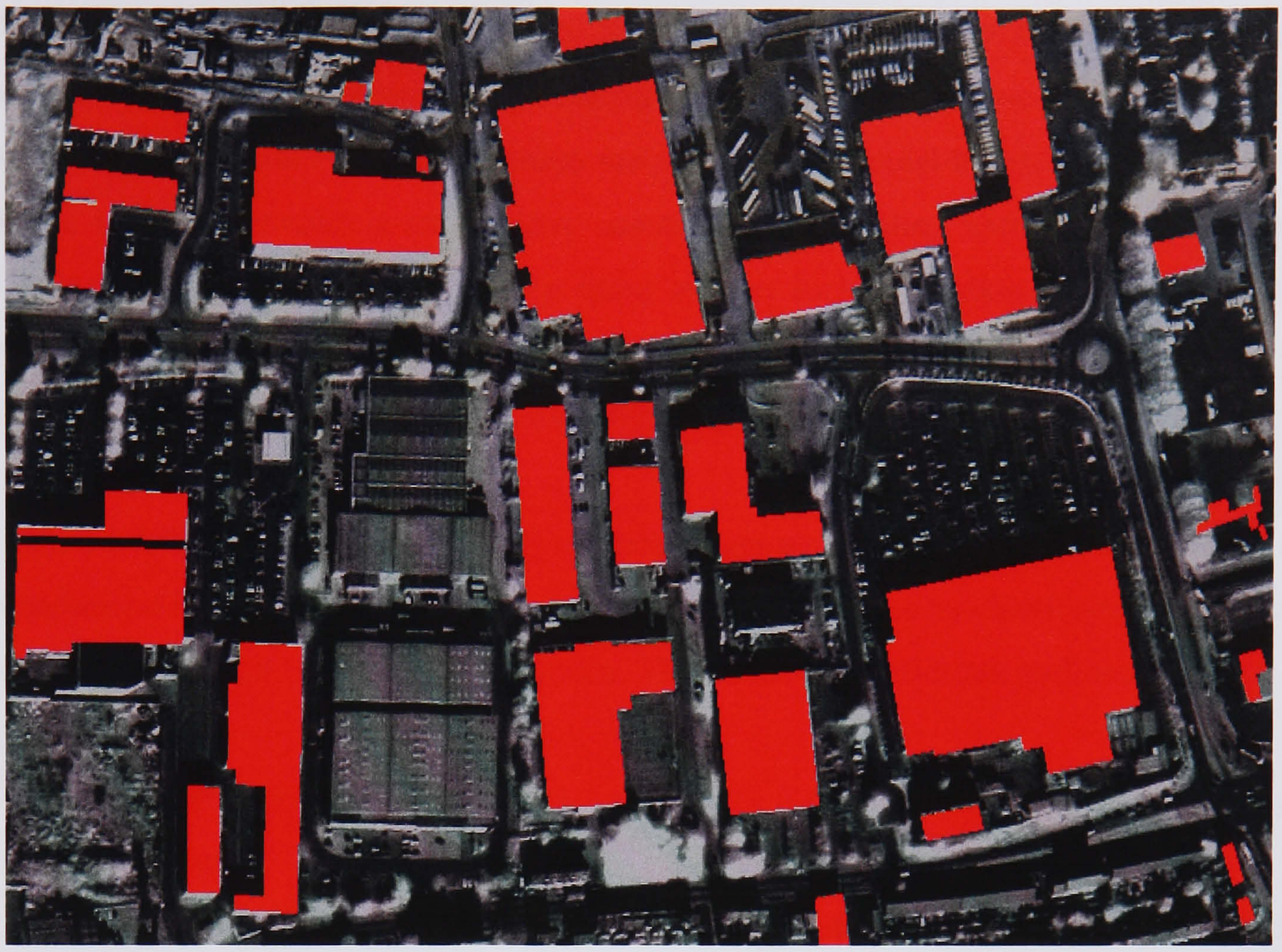


Figure 6-31. UCL building map after removing the faults in the OS data

Image removed due to third party copyright

Figure 6-32. OS MaterMap® building polygons after removing small features

A metric was required that evaluates the positional accuracy of the UCL building polygons in accurately representing building objects in the correct location. Shufelt and McKeown (1993) introduced evaluation metrics that use a pixel-by-pixel comparison in image space to provide a quantitative assessment of building extraction algorithms. These metrics have been employed by many authors such as Fradkin et al. (2001), Lee et al. (2003), Parish (2002), and Shufelt (1999). In this study, these metrics were adopted to evaluate the overall success of the proposed building extraction algorithm.

Once the UCL building map and OS reference data were co-registered to each other, a comparison between two data sets is performed in image space by classifying each pixel of both datasets into one of four categories. These four categories are outlined below:

- **True Positive (TP):** Both data sets classify a pixel as being part of a building object.
- **True Negative (TN):** Both data sets classify a pixel as part of the background, which is not relevant to building objects.
- **False Positive (FP):** UCL building map classifies a pixel as building object, while OS data set classifies it as background.
- **False Negative (FN):** UCL building map classifies a pixel as background, while OS data set classifies it as an object.

Based on the result of the pixel classification noted above, a number of objective metrics can be calculated by counting the number of pixels in each of the categories. These metrics can be defined as follows:

$$building\ \ detection\ \ percentage\ (\%) = \frac{TP}{TP + FN} \times 100 \tag{6-14}$$

$$branching\ \ factor = \frac{FP}{TP} \tag{6-15}$$

$$quality\ \ percentage\ (\%) = \frac{TP}{TP + FP + FN} \times 100 \tag{6-16}$$

Shufelt (1999) noted that each metric mentioned above provides its own quantitative measure for evaluating the overall success of building extraction algorithms. The

building detection percentage measures the fraction of the OS reference polygons that were correctly classified as building pixels in the UCL building map. It is often described as a measure of building detection performance. The *branching factor* is a measure of the degree to which the UCL building map over-classifies background pixels as building pixels. If the UCL data never "over-delineates" the extent of any building, then the branching factor would be 0, whereas a branching factor would be 1 if the UCL building map incorrectly labels a background pixel as a building pixel for every building pixel it correctly detected. Finally, the *quality percentage* measures the absolute quality of the UCL building map with respect to the OS reference data. To obtain 100% quality, the UCL building map must correctly label every building object pixel, without missing any building pixel ($FN = 0$) and without mis-labelling any background pixels ($FP = 0$).

The building detection percentage can be treated as a measure of object detection performance; the branching factor can be treated as a measure of delineation performance. The quality percentage combines aspects of both measures to summarize system performance. All three measures taken together give a clear picture of system performance with simple and unambiguous interpretations and without any subjective elements (Shufelt, 1999).

The evaluation metrics mentioned above were calculated for the UCL building map of figure 6-31 in the relation to the OS building map of figure 6-32. The results are tabulated below.

Table 6-8. Pixel classification results

Pixel classification	Pixels
True Positive (TN)	67085
True Negative (TN)	255794
False Positive (FP)	4344
False Negative (FN)	14639

Table 6-9. Building extraction metric result

Building extraction metric	Evaluation result
Building detection percentage	93.92 (%)
Branching factor	0.22
Quality percentage	77.94 (%)

As can be seen in table 6-9, the suggested building extraction technique detected building objects with 94 % detection rate (*building detection percentage*), and showed 0.2 delineation performance (*branching factor*). Finally, the overall success of the technique was evaluated as 78 % extraction quality (*quality percentage*). It is worthwhile to provide some comparison of the results reported in literature to put the success of the proposed technique into context. It is, however, difficult to make direct comparison since the evaluation metrics have been applied in only a few previous research reports. Furthermore, the specific definitions of the evaluation metrics used in the previous works are varied and different primary data have been used for extracting buildings. Nevertheless, to provide some comparison, table 6-10 summarizes the quality assessment results of building extraction systems using various primary datasets reported by many researchers.

Table 6-10. Quality evaluation results of building extraction systems in literature

System	Building detection rate (%)	Branching factor	Quality (%)	Primary data	
				Image	3D elevation
Lee et al. (2003)	64.4 – 72.1	0.4 – 2.0	29.8 -51.3	IKONOS image (single + colour)	-
Kim & Muller (2002)	85.9	1.0	45.4	IKONOS mage (single + colour)	LIDAR DSM
Fradkin et al. (2001)	50.0 - 64.1	0.1 – 0.4	49.0 - 63.6	Airborne image (single with 10cm GSM+ monochromic)	Airborne DSM

Lee et al. (2003) achieved a *building detection rate* of 64 %, a *branching factor* of 0.4 and a *quality percentage* of 51 %. Similarly, Parish (2002) evaluated the performance of the building extraction system developed by Kim & Muller (2002), which showed a *building detection rate* of 86 %, a *branching factor* of 1.0 and a *quality percentage* of 45 %. With a high-resolution airborne image and a high-quality elevation model, Fradkin et al. (2001) reported that they achieved a *building detection rate* of 64% and a *quality percentage* of 64%, which they state as being low in comparison to other techniques. It may, therefore, be valid to assume that the *building detection rate* (94 %), *branching factor* (0.2), and *quality percentage* (78 %) achieved by the suggested technique is acceptable.

Although the developed building extraction system shows a satisfactory result, it produces certain number of building extraction errors (*FP* and *FN*), which should be reduced to achieve a more accurate extraction of building objects. The errors apparent in the result generated by the technique can generally be divided into three categories;

- ***Building detection error***: these are errors that are present when building objects are not detected by the building detection process, and thus the BUS organization process cannot be applied to them.
- ***Building delineation error***: these are errors that are present when boundaries of building objects are not properly extracted by the BUS organization process.
- ***Reference data error***: these are errors that are present due to the inherent faults in the OS building polygons.

Figure 6-33 shows the building extraction errors over the Greenwich datasets, which are differently coloured in relation to error sources mentioned above. In this figure, blue-coloured pixels are errors which are caused by the building detection process. Over those buildings with long and narrow structure such as a row of terraced houses in the right-bottom of figure 6-33, the building detection process failed to group coarsely located LIDAR points as a single object so that those buildings were not successfully detected. Most of false negative (*FN*) pixels in the evaluation metrics are generated by under-detection of the terraced houses. This problem is mainly caused by the fact that the NDVI classification described in §6.4.3.2 tends to over-remove "building" points over the terraced houses and results in a very small "blob", whose member points are

less than 30 points (cf. figure 6-9 (b) and figure 6-17). This problem could be resolved by modifying the NDVI from point-wise classification to region-wise classification. In order to ensure larger numbers of member points are obtained, "high-rise" points populated by the "intermediate" classification are clustered in a number of single objects, and then a cluster-by-cluster tree detection is made by the NDVI classification. This modification may make terraced houses detectable since more member points are retained.

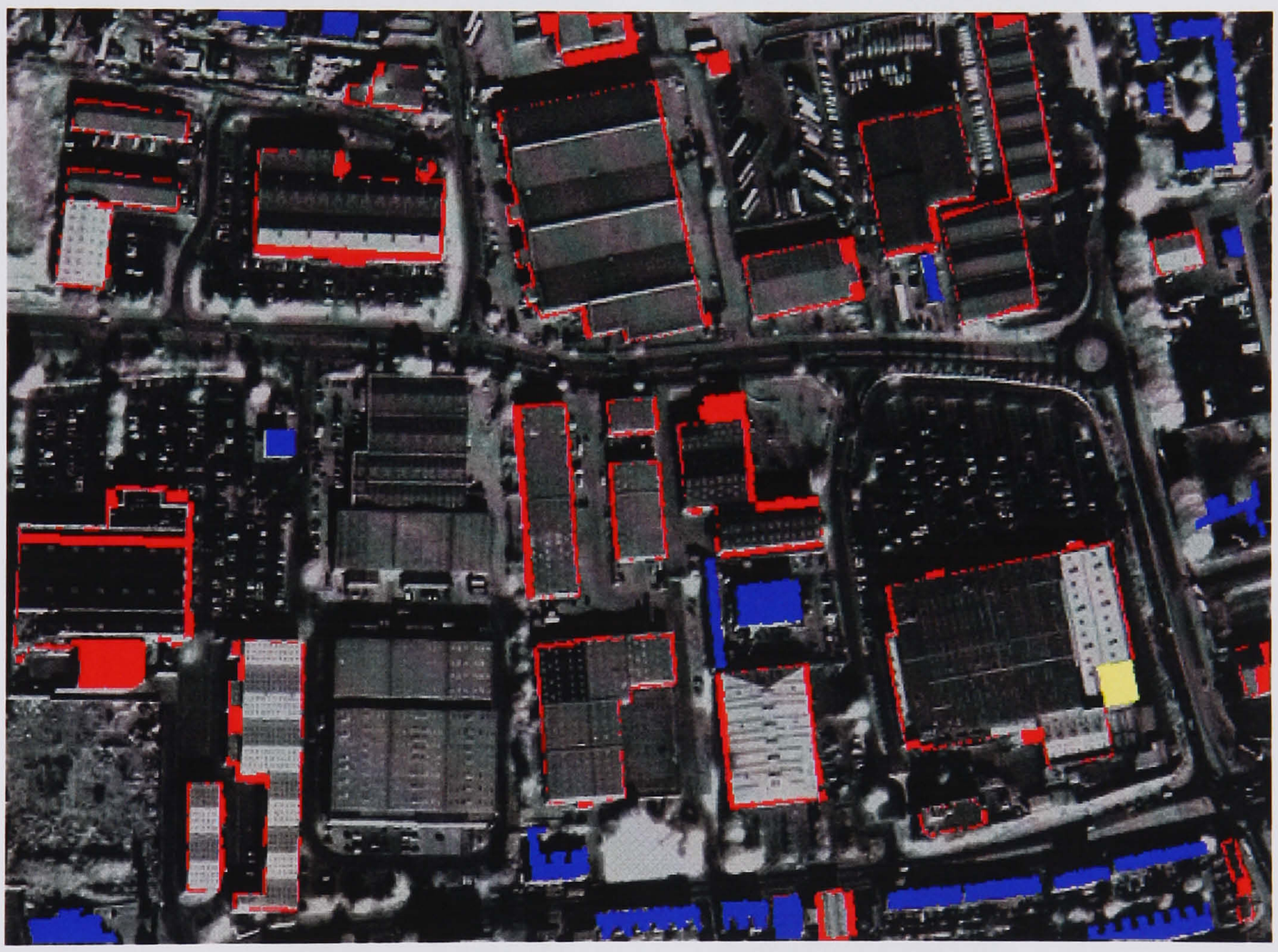


Figure 6-33. Building extraction errors

Red-coloured pixels in figure 6-33 represent errors which are generated by BUS organization process and show the delineation quality of building outlines. Those errors are related to the inherent planimetric accuracy of input data (i.e., IKONOS image, LIDAR data, and OS reference data), and the point density of LIDAR data. Most of boundary delineation errors are deviated from the OS reference data with one or two pixels if LIDAR measurements are sufficiently acquired over buildings (see figure 6-34

(a)). However, as LIDAR points are acquired with less point density over a building, more errors are produced around its boundaries (figure 6-34 (b) and (c)). This is caused by the fact that the detection of "intensity" line and "virtual" line is more difficult over a building with coarser point density than the one with denser point density. As a result, mis-location of the "intensity" line or "virtual" lines leads to the generation of delineation errors around building boundaries.

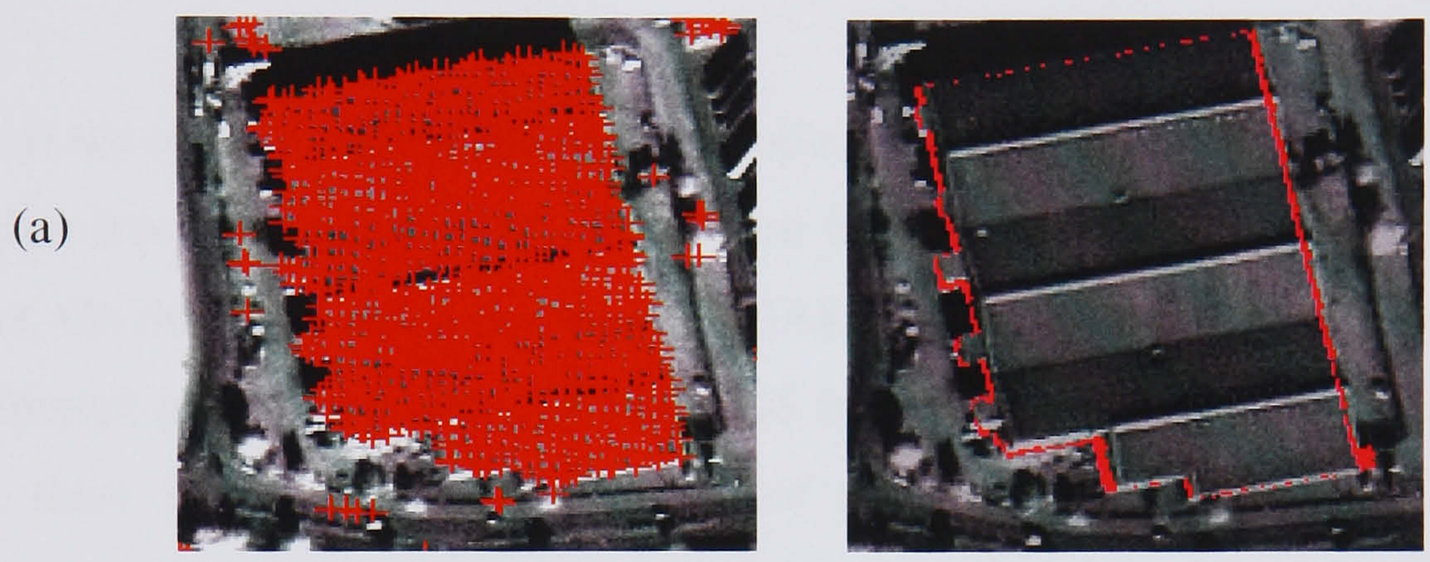
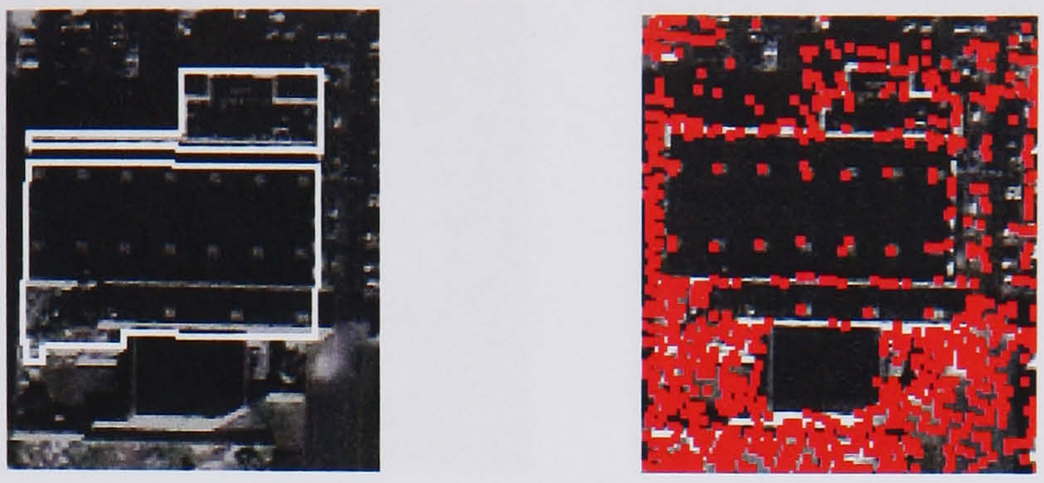


Figure 6-34. Building delineation errors; the first column shows cut-out IKONOS images overlaid with building-labelled LIDAR points; the second column shows building delineation errors

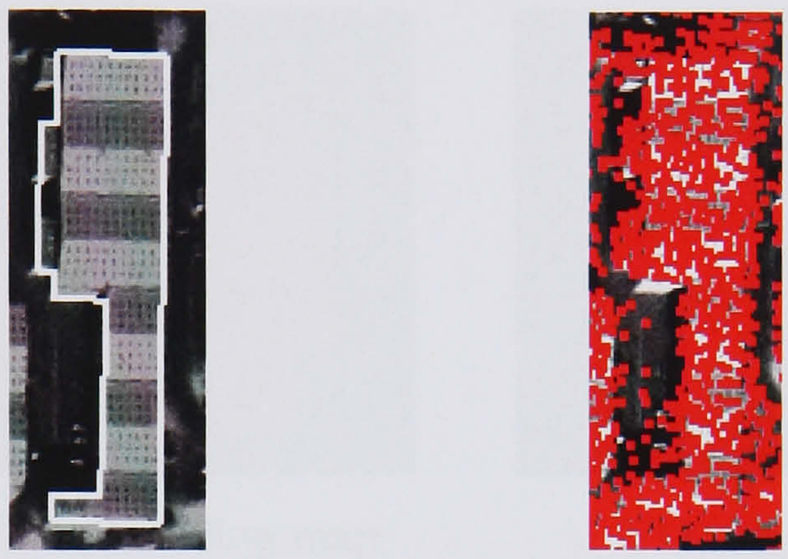
As discussed in §6.3.1, the Greenwich LIDAR DSM is acquired over some buildings with extremely low density. In this case, building shapes reconstructed by BUS organization are intruded or extruded to some extent from real building shapes or in the worst case split into two separate objects. Figure 6-35 (a) shows a example where

LIDAR points are acquired very coarsely over a building rooftop, and even no point is acquired over a lower part of that building. In this case, polygons generated over the building might be classified into "pseudo-closed" polygons, and trigger the recursive partition over them, which ends up with the generation of "empty" polygons. Thus, original building shapes are intruded as by removing such "empty" polygons. This error could be avoided if LIDAR data with higher density can be used or LIDAR data is evenly distributed so that parameters for defining the "pseudo-closed polygon" can be regularized.

There is the reverse case of the former problem, in which some building boundaries are extruded from original features (see figure 6-35 (b)). This problem is caused by two factors. On the one hand, a portion of LIDAR points is located outside buildings due to the inherent planimetric errors of IKONOS image and LIDAR data used. On the other hand, there is a significant deficiency of non-building points on the ground near buildings. When the both factors are coherently combined with each other, erroneous "closed" polygons are generated, which could be recognized as the "building" polygon, and make reconstructed building shapes extruded from their original shapes. Since non-building points are used as important evidence to verify the "building" polygon, this verification error can be avoided if more LIDAR points can be acquired on the ground near to buildings.



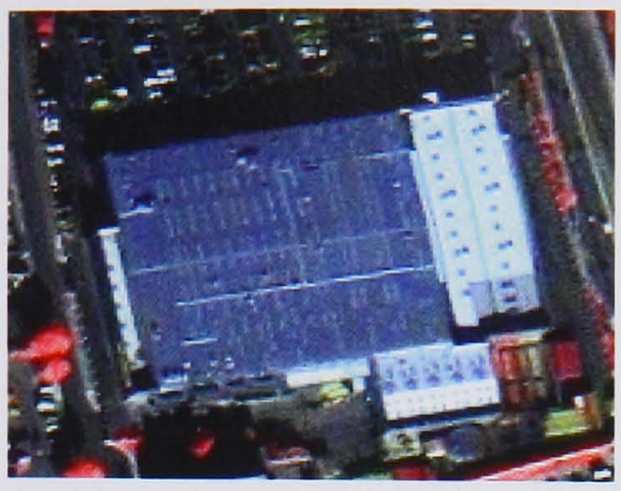
(a) intrusion of building shape



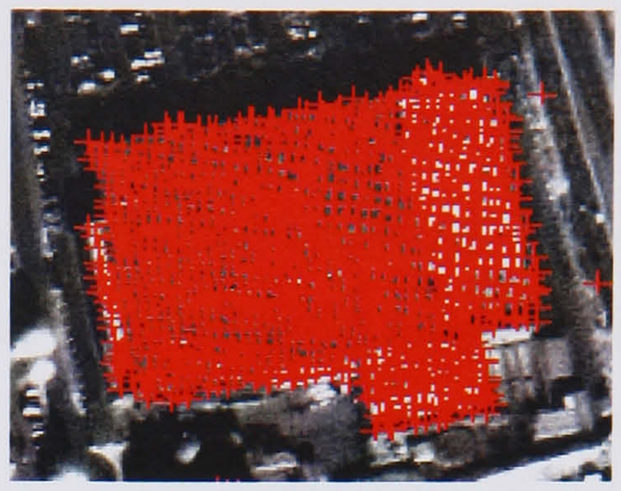
(b) extrusion of building shape

Figure 6-35. Building delineation errors caused by low point density and building outliers; white line represents building boundaries and red dot represents LIDAR points

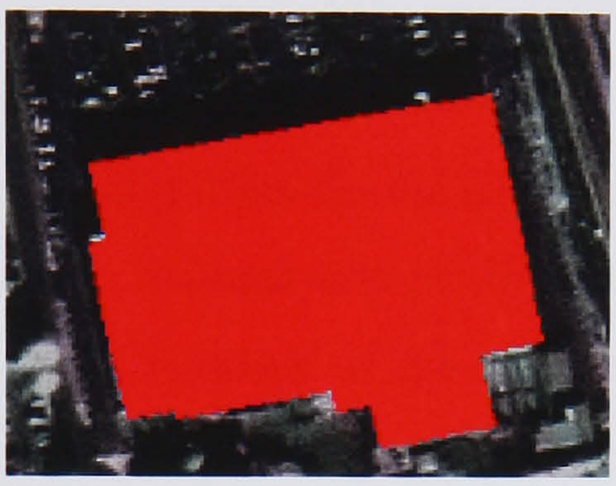
Finally, figure 6-36 shows an example of the reference data errors, which represent significant differences between the UCL building map and the OS building map. Although these errors were removed from the UCL building map before performing the quality assessment, the inherent faults in the OS data can be still found. As can be seen in figure 6-36, the UCL building map can successfully delineate boundaries of a building based on the result of LIDAR measurements and IKONOS image. However, the OS building map missed some part of that building which is coloured by yellow (figure 6-36 (e)). This is because the OS data was constructed at a different time to the acquisition of the IKONOS image and LIDAR data, from which the UCL data was generated. The analysis of the reference errors suggests that the developed building extraction technique can be also used for applications detecting changes in an urban environment and supporting map compilation.



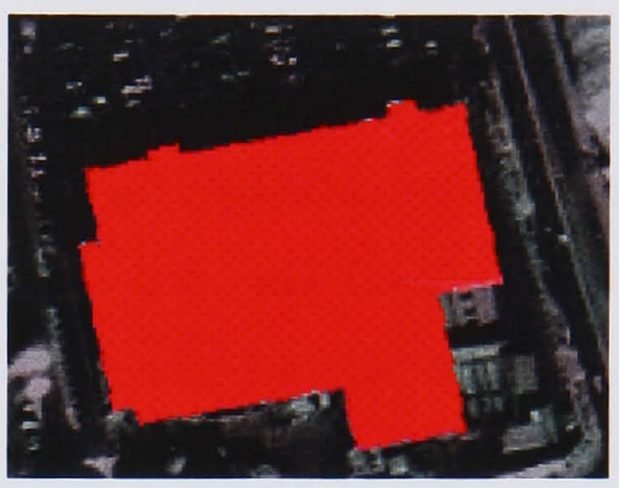
(a) IKONOS image



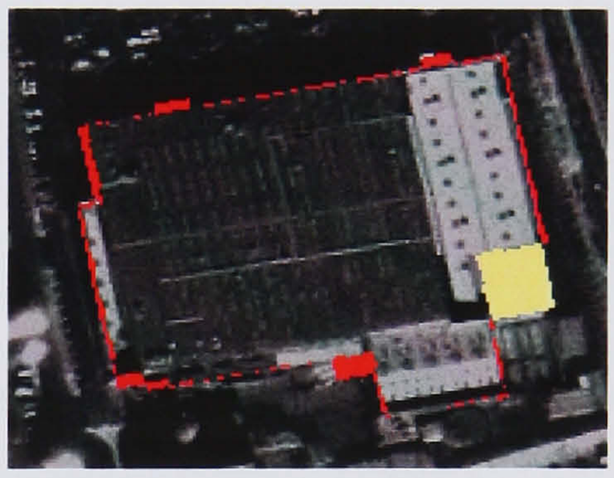
(b) LIDAR points



(c) UCL building map



(d) OS building map



(e) Reference data errors

Figure 6-36. Reference data errors

6.7 Summary

This chapter presented a new building extraction method which automatically detects building objects and delineates their boundaries using IKONOS image and LIDAR data in coherent collaboration of two different datasets. The developed technique is formulated into a two step procedure: building detection and building description.

In §6.4, a building detection method was introduced that reduces a scene complexity in urban areas and simplifies the building description process. The technique subsequently detects dominant features comprising the urban scene, and finally isolates buildings from surrounding features in the IKONOS image and LIDAR data. The RTF filter described in chapter 4 automatically reconstructs terrain surfaces from a cloud of LIDAR measurements, thereby detecting off-terrain points that are located above the terrain. Then, outlying off-terrain points, whose heights are less than a pre-defined threshold from the generated terrain surface, are labelled as low-rise points, otherwise high-rise points.

Two algorithms were suggested for detecting vegetated points from high-rise points. Although the analysis of local surface variances has been considered as a useful tool for recognizing trees, some experiments using real data showed that the method is sensitive to the point density and irregularity of LIDAR measurements. Thus, the normalized difference vegetation indices (NDVI) calculated from IKONOS colour bands was preferred as an alternative solution for tree detection. Once the trees are removed from high-rise points, the remaining points form isolated building regions (called building "blobs) and are localized with rectangles, which represent initial building shapes.

In §6.5, a building description process was presented that reconstructs building boundaries from each building "blob" resulting from the building detection process. The building object is generally represented as a mosaic of convex polygons. As a primary cue, linear features are extracted from IKONOS image, and boundary lines (called "intensity" line cues) are detected based on the classification result of LIDAR points. By employing a box-type building model, more linear features (called "virtual" line cues) are generated using LIDAR data.

Based on the BSP (Binary Space Partitioning) tree algorithm, the BUS (Building Unit Shpae) organization method was developed to generate convex polygons by a recursive intersection of linear cues, and collect relevant polygons comprising the building object. A partition scoring function was proposed to make polygonal cues generated in a level-of-detail (LoD) strategy. The generated polygons are verified as building parts by investigating the density and labelling attributes of member points. Finally, building boundaries are reconstructed by merging the verified polygons.

In §6.6, the developed building extraction system was applied to a sub-scene of the Greenwich area. The results showed that the suggested system can successfully delineate most buildings in the scene. The overall quality of the results was assessed by Shufelt & McKeown (1993)'s evaluation metrics in comparison to the OS MasterMap®. This assessment determined the building detection percentage to be 93 %, the branching factor to be 0.2, and the quality percentage to be 78 %, which in relation to other studies, is commendable. A comprehensive error analysis showed that the extracted building polygons tend to overlook some buildings (*False Negative* pixels) and misclassify some objects as buildings (*False Positive* pixels) because of poor point density of the Greenwich LIDAR DSM. These errors could be reduced if LIDAR measurements used were more evenly distributed with higher density. Furthermore, the error analysis demonstrated that the technique can automatically find the inherent difference between the new data and the OS MasterMap®, which may be a useful tool for improving a laborious topographic mapping procedure by highlighting problematic areas.

CHAPTER 7
CONCLUSIONS

7.1 Introduction

With the recent advent of a series of commercialized high-resolution satellite, the potential of IKONOS imagery in topographic mapping has been investigated and highlighted by many researchers (Holland & Marshall, 2003). However, success of fully automated reconstruction of building objects from the IKONOS imagery is still far to reach, and only partial solutions and limited success in constrained environments have been reported.

A new approach to automatically generate building boundaries from IKONOS images and LIDAR data has been developed. It was shown that acquiring accurate detection and description of building objects from remotely sensed data is fundamental for a variety of applications, and still requires detailed research. Some difficulties which hinder the automation of the building extraction in an urban environment were identified and described in terms of scene complexity, lack of building modelling cues, and complexity of integrating multiple sensor data. In order to circumvent these problems, a method of automatic building extraction was proposed that is accomplished by two independent tasks; *building detection* and *building description*. The success of this method was evaluated using real datasets. The result has shown that, if the complementary nature of the IKONOS image and LIDAR data is combined, the automation of the building extraction can be greatly improved and conventional compilation of topographic maps can be accomplished in much less time and with very little human effort by support of this method. Moreover, the method can detect terrain surfaces, low-rise objects, and trees as well as buildings, which may be considered as a

partial solution to generate the urban landscape model (ULM). This chapter will summarize how the proposed method of automatic building extraction has been developed in this thesis. Then, it will outline where the weaknesses in the study are, and how these weaknesses could be overcome in future work. Finally, this chapter will summarize research achievements of this study.

7.2 Automatic building detection

Due to a high scene complexity in urban scenes, a very large number of geometric features, which are not relevant to building objects, are normally produced. It is hard to determine whether a feature belongs to a building object since other objects such as roads, trees, and nearby buildings have similar geometric characteristics.

An automatic building detection method was proposed to efficiently reduce a high scene complexity and simplify the building extraction process. The developed method aims to separate single building from its surrounding features, and localize it by using IKONOS image and LIDAR data. By that, the building description process can be focused on an isolated building region, and plausible building extraction errors can be removed in advance. This task was achieved by sequentially removing dominant urban features including terrain surfaces, low-rise objects and trees. This hierarchical building detection strategy was originally adopted form the work of Baillard & Maitre (1999). However, the representation of each object was uniquely defined, and new algorithms to automatically detect them were suggested.

In chapter 3, a new filtering algorithm, called RTF (Recursive Terrain Fragmentation) filter was proposed to detect terrain points from a cloud of LIDAR measurements, thereby obtaining non-terrain points. The major problem of the LIDAR filtering techniques was identified as a lack of generic knowledge about the terrain surface and non-terrain objects. Generic terrain surfaces are generally represented as a mosaic of planar terrain surfaces. Based on this, the proposed filter can explicitly model the terrain surfaces with minimal use of parameters, and efficiently cope with a variety of terrain slopes.

The RTF filter was implemented by employing a hypothesis-test optimization in different scales. The entire LIDAR space, initially hypothesized with a single planar

terrain surface, was recursively fragmented with small sub-regions until the coexistence of different terrain slopes cannot be found over all fragmented regions. This recursive fragmentation was achieved by hypothesizing a given area as a planar terrain surface by a number of model candidates and selecting the "best" model. By investigating contextual labelling attributes of LIDAR points, an optimum criterion to select "better" planar terrain model amongst model candidates was proposed in terms of "terrain polarity", and implemented using MDL (Minimum Description Length) framework. A few new ideas to correct filtering errors were also developed by reasoning contextual labelling attributes of consecutively connected LIDAR points.

In chapter 4, the overall success of the RTF filter was evaluated using two full scenes of LIDAR data, each of which has different terrain characteristics. The filter was applied to the Shrewsbury data. The quality of filtering results was assessed in comparison with various reference datasets. The results showed that the filter can successfully remove most of non-terrain features, while well preserve morphological terrain shapes. Compared to the OS Landline.Plus® building vectors, it was confirmed that most of filtered terrain points were located outside of the vectors. Also, the vertical accuracy of the filtered terrain surface was measured as 0.15 m of RMS residuals in comparison with the KPGS profiles.

During the experiment, the RTF filter has been modified to more remove low objects. It has been achieved by selecting different filtering criterion according to a minimum size of underlying local plane terrain model. This adaptive strategy makes the filter focus on the detection of terrain slope variations over larger areas, whereas it is focused on the detection of low objects over smaller areas. Some additional improvements of the RTF were implemented to make the filter more adaptive to a high terrain complexity in urban environments. This has been implemented by classifying discontinuous terrain surfaces into more detail terrain classes, and thus applying different filtering algorithms according to the terrain classification result. The modification of the RTF filter was shown to be successful at further removing low objects by 9.4 % from the Shrewsbury data.

The modified RTF filter was applied to a complex urban scene using the OEEPE LIDAR DSM. In comparison with manually filtered reference data, the filtering errors

of the RTF filter was measured as 0.94% for Type I ("omission") error, whereas 6.75 % for Type II (commission) error. A comprehensive analysis has demonstrated that those filtering errors did not significantly degrade the overall quality of the terrain model reconstructed, and were limited to low objects less than 1 m in height from the terrain surface.

Chapter 6 proposed a sequential processing chain for the building detection. Once the terrain surface was reconstructed by the RTF filter, outlying points with a height larger than a pre-defined threshold from the generated terrain surface were classified as "high-rise" points. Two algorithms were suggested for detecting tree points from the high-rise points. Considering the rather poor quality of the LIDAR data used, it was preferred to use the normalized difference vegetation indices (NDVI) calculated from IKONOS colour bands for the tree detection, instead of the method relying on the analysis of local surface variances. After removing the tree points, the remaining points formed isolated building regions (called building "blobs"). The proposed building detection process was terminated as localizing the detected building "blobs" with rectangles, which represent initial building shapes.

7.3 Automatic building description

It is difficult task to extract sufficient cues to describe a complete object shape from remotely sensed data. This is because information acquired by the sensors is normally disrupted by several factors such as the illumination condition, resolution effect, object complexity and disadvantageous image perspective. This study proposed a building description method to recover a full delineation of building boundaries from incomplete geometric features extracted from the remotely sensed data.

Chapter 5 proposed a method of building description to delineate building boundaries from monocular IKONOS imagery. Generic knowledge about the shape of the building object was represented as a mosaic of convex polygons. Compared to specific building models, the suggested representation was considered to be better suited for describing complex shapes of buildings. A global analysis of dominant building orientation based on an analytic function of Fourier transformation was suggested to remove distracting features which are not relevant to building boundaries. A BUS (Building Unit Shape) organization method was introduced as a middle-level of feature inferring and grouping solution, which can make highly fragmented or completely missed lines recovered, and

finally reconstruct building outlines. The method segmented a manually localized building region into polygons by recursive intersections of lines extracted. Similar to the region-growing technique, polygons with comparable intensity values were merged so that spurious lines were eliminated and finally building boundaries were reconstructed. The developed technique was applied to a sub-scene of the Farnborough IKONOS imagery. The implication of the results is that the method can overcome the shortcomings of conventional area-based segmentation for the building description, and some drawbacks of the method need further improvement.

In chapter 6, the proposed method has been improved in order to achieve a full automation and higher level of geometric freedom for delineating building boundaries by using LIDAR data as an additional primary data. The method was developed in a fusion framework, in which data-driven and model-driven linear features were combined in order to increase cue density, and the weaknesses of individual cues extracted from IKONQS image and LIDAR data were compensated for by others.

The method was applied to each building "blob" resulted from the building detection process to reconstruct building boundaries. At a cue extraction stage, linear features were extracted from IKONOS image, which were then verified as boundary lines based on the classification result of LIDAR points. By employing a box-type building model, more linear features were generated using LIDAR data.

The BUS organization method has been modified in consideration of the contribution of LIDAR data. Within a localised building "blob", the method generated convex polygons by a recursive intersection of linear cues, and collect relevant polygons comprising the building object. A partition scoring function was proposed to make polygonal cues generated in a level-of-detail (LoD) strategy. The generated polygons were then verified as building parts by investigating the density and labelling attributes of member points. Finally, building boundaries were reconstructed by merging the verified polygons.

Using the Greenwich IKONOS image and LIDAR data, the success of the proposed method was evaluated by Shufelt & McKeown (1993)'s evaluation metrics in comparison to the OS MasterMap®. The result showed that the method can extract most of buildings in the scene with 93 % of the building detection percentage, 0.2 of the

branching factor, and 78 % the quality percentage, which in relation to other studies, is commendable. Furthermore, it has demonstrated that the technique can automatically find the inherent faults of the OS MasterMap®, which may be used for change detection applications.

7.4 Future research work

It has demonstrated that the RTF filter can reconstruct terrain surfaces from LIDAR data with a level of accuracy, which is well suited for accomplishing the building detection. However, it was found that the filter over-classified low-objects and the Type II errors ("commission" errors) were still highly produced. More accurate terrain surface would be required for other application such as environmental risk assessment. Thus, those filtering errors must be reduced. An adaptive strategy to select different filtering criterion for removing low-objects has been suggested in chapter 4. The problem of this method can be ascribed as lack of certainty to define the relationship between the height of the low-object and its physical extent, which leads to the difficulty to select filtering parameters. This problem could be resolved by employing more reliable representation of the low-object. In general, the low-objects comprise of vegetation, vehicles, and street furniture. It may be reasonable to assume that man-made low-objects are generally much smaller than buildings, and the surface roughness of natural low-objects is higher than the one of building rooftops. As suggested by Filin (2002), clustering LIDAR points into pre-determined feature classes becomes practical. Similar clustering algorithm can be involved in the RTF filter as a pre-processing module, and segment LIDAR points into a number of groups by merging similar surface roughness. By simply thresholding those clusters according to their size and surface roughness measure, problematic man-made and natural low-objects can be removed in advance, and the Type II errors could be reduced. The measurement of the surface roughness and its regional analysis has been suggested in §6.4.3.1 for the tree detection. This method could be applied for clustering the low-objects.

The study has illustrated that more reliable detection of the tree object can produce higher quality of the building detection. Two algorithms were proposed in §6.4.3.1 and §6.4.3.2, each of which relies on surface variance or chromatic property as primary cue for the tree detection. However, both algorithms might be problematic when LIDAR points are not sufficiently acquired; or the NDVI is wrongly computed due to

disadvantageous image perspective and illuminating condition; or the surface material of a rooftop contains the "greenness" similar to nature vegetation. Since IKONOS image and LIDAR data naturally have the complementary properties of each other, it would improve the building detection rate if both datasets are combined to detect and verify the tree object. For instance, trees detected by the NDVI classification can be verified by the surface roughness, or vice versa.

Although the proposed method of automatic building description has proved successful, there are number of areas that require further research. In particular, most of building extraction errors came from under-reconstruction of terraced houses. For achieving a full automation of topographic mapping, those objects must be reconstructed. This problem is mainly caused by the fact that the NDVI classification for the tree detection tends to over-remove "building" points over terraced houses. As a result, terraced houses were not properly localized so that the building description process cannot be applied to those objects. This problem could be resolved by modifying the NDVI method from point-wise classification to region-wise classification. That is, once "high-rise" points are populated, those points are grouped in a number of single objects in order to ensure a large number of LIDAR points can be obtained. Then, the NDVI classification serves to determine whether or not those clusters can be recognized as the tree objects. This solution may make terrace houses detectable since more member points are retained. If necessary, the NDVI can further remove tree points which are located nearby buildings relying on the point-wise approach.

The proposed method of automatic building description can be easily expanded to achieve broader applications. Firstly, the current system employed only box-type building model to generate the "virtual" lines. However, other parametric building models including parabolic model can be accommodated. This modification can make the system delineate more complicate building shapes including curvature-like shape. Secondly, the proposed method can be expanded to utilize a range of different sensors as primary data. For instance, the method can be applied to monocular image as illustrated in chapter 5. Although the suggested technique using monocular IKONOS image has a distracting feature to require manual detection of buildings, this problem could be automated by adopting the work of Lee et al. (2003), which highlighted the potential for the IKONOS colour information in automatic building detection

application. Furthermore, it would be an interesting research topic for further development of the proposed technique to replace the IKONOS image with higher resolution satellite imagery such as QuickBird-2 (~0.5 m GSD) or airborne imagery (less than 0.5 m GSD) or to use LIDAR data with higher point density (higher than 1 point per square metre).

There are, however, a number of research issues that must be overcome in order to develop broader applications for the proposed building extraction system. These issues can be described as follows:

- The developed system prefers to have an image which is acquired with nadir view and is accurately geo-referenced. However, such a high-quality image is not always available. Furthermore, LIDAR systems often produce a certain amount of geo-referencing errors, which are difficult to correct by the calibration procedure as discussed in §2.3. Therefore, techniques for supporting coherent co-registration of optical images with perspective views and LIDAR data must be provided.
- The selection of parameters to control a level-of-description (LoD) of building shapes are subject to the point density and irregularity of LIDAR measurement as discussed in chapter 6. Since a range of LIDAR systems which provide different point densities and various qualities are now available, it is necessary to give the end users better understanding how to select those parameters. To this end, a more rigorous analysis of the relationship between the number of directionality of hyperlines extracted (i.e., LoD of building shapes required) and the quality of LIDAR data must be provided.
- For other applications such as urban planning, it would be desirable to reconstruct 3D roof structures as well as building outlines. The proposed method may be expanded for the reconstruction of 3D roof structures. For instance, a 3D building roof structure could be reconstructed if building boundaries produced are re-partitioned by linear features, but this time which are located over rooftops, and to achieve maximal member points with similar planar surface characteristic.

7.5 Summary of achievements

A lot of work has been done in the field of automatic building extraction. In spite of considerable research effort, a complete system, which can be applied to a variety of scene domains and produce reliable results, has not been achieved yet and is urgently required. In view of this, the contribution of this research can be described as follows:

1. This study identified the potential of IKONOS imagery and LIDAR data in topographic mapping and actively exploited it to construct a system of automatic building extraction by maximizing the synergy of both datasets. The major achievement of the research is to devise a new technique for the extraction of buildings using a number of modules, most of which have been developed by the author and described in this thesis.
2. Although the suggested method has been originally developed to reconstruct geometric building boundaries, the very nature of the technique can also generate the ULM (Urban Landscape Model) in 3D by detecting the other dominant urban features such as terrain surfaces, low-rise objects and trees.
3. The research provided a new LIDAR filter which automatically reconstructs generic terrain surfaces from a cloud of LIDAR points. The filter can be dynamically adaptive to highly complex urban terrains with minimal use of constraints.
4. The research provided a new building model generation tool, called the BUS organization method, which give a high-level of geometric freedom for delineating building boundaries by optimally combining both generic and specific building models.
5. The proposed method of automatic building description introduced a few new ideas to generalise the shapes of building objects according to the level of details required at different map scales.

REFERENCES

Ackermann, F., 1999. Airborne laser scanning present status and future expectation. *ISPRS Journal of Photogrammetry & Remote Sensing*, 54:64-67.

Adams R., and Bischof, L., 1994. Seeded region growing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 16(6):641-647.

Ameri, B. and Fritsch, D., 1999. 3-D reconstruction of polyhedral-like building models. *International Archives of Photogrammetry and Remote Sensing*, 32(3/2W5):15-20.

Ameri, B., 2000. *Automatic recognition and 3D reconstruction of buildings from digital imagery*. Ph.D. Thesis, Institute of Photogrammetry, Stuttgart University.

Ameri, B., and Fritsch, D., 2000. Feature Based Model Verification (FBMV): a new Concept for Validation in Building Reconstruction. *International Archives of Photogrammetry and Remote Sensing*, 33(B3/1):24-35.

Anderson, H.-E., Reutebuch, S.E. and Schreuder, G.F., 2002. Bayesian object recognition for the analysis of complex forest scenes in airborne laser scanner data. *Proceedings of ISPRS Commission III, Symposium 2002 on Photogrammetric Computer Vision*. Graz, Austria, 9-13 September 2002, 34(3A):35-41.

Axelsson, P., 1999. Processing of laser scanner data - algorithms and applications. *ISPRS Journal of Photogrammetry and Remote Sensing*, 54(2-3):138-147.

Axelsson, P., 2000. DEM Generation from laser scanner data using adaptive TIN Models. *International Archives of Photogrammetry and Remote Sensing*, Amsterdam, The Netherlands, 32(B4/1):110-117.

Axelsson, P., 2001. Ground estimation of laser data using adaptive TIN-models. *Proceedings of OEEPE workshop on airborne laserscanning and interferometric SAR for detailed digital elevation models*, Royal Institute of Technology Department of Geodesy and Photogrammetry, Stockholm, Sweden, 1-3 March 2001, paper 5 (11 pages).

Baillard, C. and Maitre H., 1999. 3D reconstruction of urban scenes from aerial stereo imagery: A focusing strategy. *Computer Vision and Image Understanding*, 76(3):244-258.

Baillard, C., Schmid, C., Zisserman, A. and Fitzgibbon, A., 1999. Automatic line matching and 3D reconstruction of buildings from multiple views. *International Archives of Photogrammetry and Remote Sensing*, Munich, Germany, 32(3/2W5):69-80.

Baillard, C. and Dissard, O., 2000. A stereo matching algorithm for urban digital elevation models. *Photogrammetric Engineering & Remote Sensing*, 66(9):1119-1128.

Baillard, C. and Zisserman, A. 2000. A plane-sweep strategy for the 3D reconstruction of buildings from multiple images. *Proceedings of ISPRS Commission Workshop, Amsterdam, 2000*. CD-ROM.

Baltsavias, E., Mason, S. and Stallman, D., 1995. Use of DTMs/DSMs and orthoimages to support building extraction. In A. Grün, O. Kübler, & P. Agouris (eds), *Automatic Extraction of Man-Made Objects from Aerial and Space Images*: 199-210, Birkhäuser-Verlag, Basel.

Baltsavias, E.P., 1999. A comparison between photogrammetry and laser scanning. *ISPRS Journal of Photogrammetry and Remote Sensing*, 54(2-3):83-94.

Baltsavias, E., Pateraki, M. and Zhang, L., 2001. Radiometric and geometric evaluation of Ikonos Geo images and their use for 3D building modelling. *Proceedings of Joint ISPRS Workshop on High Resolution Mapping from Space*, 2001, Hannover, 19-21 September 2001. CD-ROM.

Bayir, I., 2003. IKONOS collection capability with direct tasking from a regional operation center. *Proceedings of Joint ISPRS Workshop on High Resolution Mapping from Space 2003*, Hannover, Germany, 6-8 October 2003. CD-ROM.

Belli, T., Cord, M. and Jordan, M., 2001. 3D data reconstruction and modeling for urban scene analysis. In A. Grün, E.P. Baltsavias & O. Henricsson (eds), *Automatic Extraction of Man-Made Objects from Aerial and Space Images (III)*: 125-134. Balkema, Lisse, the Netherlands.

Berthod, M., Gabet, L., Giraudon, G. and Lotti, J.L., 1995. High resolution stereo for the detection of buildings. In Grün, A., Kübler, O. and Agouris, P. (eds), *Automatic Extraction of Man-Made Objects from Aerial and Space Images*: 135-144. Birkhäuser-Verlag, Basel.

Beveridege, J.R., Weiss, R. and Riseman, E.M., 1989. Optimisation of 2D model matching. *Proceedings of ARPA Image Understanding Workshop*: 815-830.

Bignone, F., Henricsson, O., Fua, P. and Stricker, M., 1996. Automatic extraction of generic house roofs from high resolution aerial imagery. *Proceedings of the European Conference on Computer Vision*, 1:85-96.

Boyer, K.L. and Sarkar, S., (eds) 2000. *Perceptual organization for artificial vision systems*. Kluwer, Boston.

Brenner, C., 1999. Interactive Modeling tools for 3D Building Reconstruction, In D. Fritsch & R. Spiller (eds), *Photogrammetric Week '99*: 23-34. Herber Wichmann Verlag, Heidelberg.

Brenner, C. and Haala, N., 1998. Fast production of virtual reality city models. *Proceedings of ISPRS Commission IV Symposium on GIS - Between Visions and Applications*, Stuttgart, Germany, 32(4):77-84.

Brenner, C., 2000. Towards fully automatic generation of city models. *International Archives of Photogrammetry and Remote Sensing*, 33(B3/1):85-92.

Briese, C., Pfeifer, N. and Dorninger, P., 2002. Applications of robust interpolation for DTM determination. *Proceedings of ISPRS Commission III, Symposium 2002 on Photogrammetric Computer Vision*, Graz, Austria, 9-13 September 2002, 34 (3A):55-61.

Brovelli, M.A., Cannata, M. and Longoni, U.M., 2002. Managing and processing LIDAR data within GRASS. *Proceedings of the GRASS Users Conference 2002*, Trento, 11-13 September 2002.

Brunn, A. and Weidner, U., 1997. Extracting buildings from digital surface models. *International Archives of Photogrammetry and Remote Sensing*, 32(3/4W2):27-34.

Burman, H., 2000. Adjustment of laserscanner data for correction of orientation errors. *International Archives of Photogrammetry and Remote Sensing*, 33(B3/1):125-132.

Burns, J.B., Hanson, A.R. and Riseman, E.M., 1986. Extracting straight lines. *IEEE Pattern Analysis and Machine Intelligence*, 8(4):425-455.

Canny, J.F., 1986. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679-698.

Chabat, F., Yang, G.Z. and Hansell, D.M., 1999. A corner orientation detector. *Image and Vision Computing*, 17(10):761-769.

Cham, T.J. and Maitre, H., 1999. Automated B-spline curve representation incorporation MDL and error-minimizing control point insertion strategies. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(1):49-53.

Cord, C., Jordan, M. and Cocquerez, J.-P. and Paparoditis, N., 1999. Automatic extraction and modelling of urban buildings from high resolution aerial images. *International Archives of Photogrammetry and Remote Sensing*, München, Germany, September, 32(3/2W5):187-192.

Cord, M., Jordan, M. and Cocquerez, J.–P., 2001. Accurate building structure recovery from high resolution aerial imagery. *Computer Vision and Image Understanding*, 82:138-173.

Crombaghs, M., Brügelmann, R. and De Min, R., 2000. On the adjustment of overlapping strips of laseraltimeter height data. *International Archives of Photogrammetry and Remote Sensing*, 33(B3/1):224-231.

CVIU, 1998. Special issue on building extraction. *Computer Vision and Image Understanding*, 72(2).

Danahy, J., 1999. Visualization data needs in urban environmental planning and design. In D. Fritsch & R. Spiller (eds), *Photogrammetric Week '99*: 351-365. Herber Wichmann Verlag, Heidelberg.

Dial, G., Gibson, L. and Poulsen, R., 2001. Ikonos satellite imagery and its use in automated road extraction. In A. Grün, E.P. Baltsavias & O. Henricsson (eds),

Automatic Extraction of Man-Made Objects from Aerial and Space Images (III): 357-368. Balkema Publishers, Lisse, the Netherlands.

Dial, G. and Grodecki, J., 2002. Block adjustment with rational polynomial camera Models. *Proceedings of ASPRS 2002 Conference*, Washington, DC, 22-26 April 2002. CD-ROM.

Dial, G. and Grodecki, J., 2003. Applications of IKONOS Imagery. *Proceedings of ASPRS 2003 Conference*, Anchorage, Anchorage, Alaska, 5-9 May. CD-ROM.

Dowman, I.J., 2000. Automatic feature extraction for urban landscape models. *Proceedings of the 26th annual conference of the remote sensing society*, 12-14 September 2000. CD-ROM.

Dowman, I.J. and Dolloff, J., 2000. An evaluation of rational functions for photogrammetric restitution. *International Archives of Photogrammetry and Remote Sensing*, 33(B3):254-266.

Dowman I.J. and Tao, C.V., 2002. An update on the use of rational functions for photogrammetric restitution. *ISPRS Highlights*, 7(3):22-29.

Dold, C., 2001. Filtering of laser altimetry data – conventional methods and a new adaptive approach realized with a graphical user interface. M.Sc. thesis, University College London, London.

Elberink, S.O., Brand, G., and Brügelmann, R., 2003. Quality improvement of laser altimetry DEMs. *Proceedings of ISPRS working group III/3 workshop on 3-D reconstruction from airborne laserscanner and InSAR data*, Dresden, Genrmany, 8-10 October 2003, 34(3/W13):51-58.

Elmqvist, M., 2001. Ground estimation of laser radar data using active shape models. *Proceedings of the OEEPE workshop on airborne laserscanning and interferometric SAR for detailed digital elevation models*, Royal Institute of Technology Department of Geodesy and Photogrammetry, Stockholm, Sweden, 1-3 March 2001, 5 pages.

Filin, S., 2002. Surface clustering from airborne laser scanning data. *Proceedings of ISPRS Commission III, Symposium 2002 on Photogrammetric Computer Vision*, Graz, Austria, 9-13 September 2002, 34 (3A):119-224.

Filin, S., 2003. Analysis and implementation of a laser strip adjustment model. *Proceedings of ISPRS working group III/3 workshop on 3-D reconstruction from airborne laserscanner and InSAR data,* Dresden, Genrmany, 8-10 October 2003, 34(3/W13):65-70.

Fischer, A., Kolbe, T., Lang, F., Cremers, A., Frstner, W. Plmer, L. and Steinhage, V., 1998. Extracting Buildings from Aerial Images Using Hierarchical Aggregation in 2D and 3D. *Computer Vision and Image Understanding*, 72(2):185-203.

Fischer, P., 2002. Investigation on the accuracy of digital elevation models from stereo-photogrammetry and airborne laser scanning. M.Sc. Thesis, University College London, London.

Flood, M. and Gutelius, B., 1997. Commercial implications of topographic terrain mapping using scanning airborne laser radar, *Photogrammetric Engineering and Remote Sensing*, 63(4):327-329 and 363-366.

Flood, M., 2001. LIDAR activities and research priorities in the commercial sector. *International Archives of Photogrammetry and Remote Sensing*, Annapolis, MD, 34(3/W4):3-7.

Fowler, R., 2001. Topographic Lidar. In D.F., David (eds), *Digital Elevation Model Technologies and Applications: The DEM Users Manual*: 207-236. ASPRS, Maryland.

Forstner, W. and Plumer, L., (eds) 1997. Semantic modelling for the acquisition of topographic information from images and maps. Birkhauser Verlag, Bonn, Germany.

Fradkin, M., Maitre, H. and Roux, M., 2001. Building detection from multiple aerial images in dense urban areas. *Computer Vision and Image Understanding,* 82:181-207.

Fraser, C.S., Baltsavias, E. and Gruen, A., 2001. 3D building reconstruction from high-resolution Ikonos stereo images. In A. Grün, E.P. Baltsavias & O. Henricsson (eds), *Automatic Extraction of Man-Made Objects from Aerial and Space Images (III)*: 331-344. Balkema Publishers, Lisse, the Netherlands.

Fraser, C., Hanley, H. and Yamakawa, T., 2002a. 3D geopositioning accuracy of IKONOS imagery. *Photogrammetric Record*, 17(99):465-479.

Fraser, C., Hanley, H. and Yamakawa, T., 2002b. High-precision geopositioning from IKONOS satellite Imagery. *Proceedings of ASCM-APRSR Annual Convention,* Washington DC, 19-26 April 2002. CD-ROM.

Fraser, C.S. and Yamakawa, T., 2003. Applicability of the affine model for Ikonos image orientation over mountainous terrain. *Proceedings of Joint ISPRS Workshop on High Resolution Mapping from Space 2003*, Hannover, 6-8 October 2003. CD-ROM.

Fuchs, H., Kedem, Z.M. and Naylor, B.F., 1980. On visible surface generation by a priori tree structures. *Computer Graphics*, 14(3):124-133.

Gabet, L., Giraudon, G. and Renourad, L. 1997. Automatic generation of high resolutin urban zone digital elevation models. *ISPRS Journal of Photogrammetry and Remote Sensing*, 52(1997):33-47.

Gambotto, J.P., 1993. A new approach to combining region growing and edge detection. *Pattern Recognition Letters*, 14(11):869-875

Gibson, L., 2003. Finding road networks in IKONOS satellite imagery. *Proceedings of ASPRS 2003 Conference*, Anchorage, Alaska, 5-9 May 2003. CD-ROM.

Grodecki, J., 2001. IKONOS stereo feature extraction - RPC approach. *Proceedings of ASPRS 2001 Conference*, St. Louis, 23-27 April 2001. CD-ROM.

Grodecki, J. and Dial, G., 2001. IKONOS geometric accuracy. *Proceedings of Joint ISPRS Workshop on High Resolution Mapping from Space 2001*, Hannover, 19-21 September 2001. CD-ROM.

Grodecki, J. and Dial, G., 2002. IKONOS geometric accuracy validation. *Proceedings of ISPRS Commission I Mid-Term Symposium*, Denver, 10-15 November. CD-ROM.

Grodecki, J. and Dial, G., 2003. Block adjustment of high-resolution satellite images described by rational polynomials. *Photogrammetric Engineering & Remote Sensing*, 69(1):59-68.

Gruen, A., Kuebler, O. and Agouris, R., (eds) 1995. *Automatic Extraction of Man-Made Objects from Aerial and Space Images*. Birkhäuser, Basel, Boston, Berlin.

Gruen, A., Baltsavias, E. and Henricsson, O., (eds) 1997. *Automatic Extraction of Man-made Objects from Aerial and Space Images (II)*, Birkäuser, Basel, Switzerland.

Haala, N., 1994. Detection of buildings by fusion of range and image data. *International Archives of Photogrammetry and Remote Sensing*, 30(3/1):341-346.

Haala N. and Hahn, M., 1995. Data fusion for the detection and reconstruction of buildings. In A. Grün, O. Kübler, & P. Agouris (eds), *Automatic Extraction of Man-Made Objects from Aerial and Space Images*: 211-220. Birkhäuser-Verlag, Basel.

Haala, N., Anders, K.-H., 1997. Acquisition of 3D urban models by analysis of aerial images, digital surface models and existing 2D building information, *Proceedings of SPIE Conference on Integrating Photogrammetric Techniques with Scene Analysis and Machine Vision III*: 212-222. Orlando, Florida.

Haala N. and Brenner C., 1998. Interpretation of urban surface models using 2D building information, *Computer Vision and Image Understanding*, 72(2):204-214.

Haarbrink, R.B., 2003. High altitude LIDAR to enhance GEOSAR system performance. *Proceedings of ISPRS working group III/3 workshop on 3-D reconstruction from airborne laserscanner and InSAR data*, Dresden, Genrmany, 8-10 October 2003, 34(3/W13):199-204.

Hanley, H. and Fraser, C., 2001. Geopositioning accuracy of Ikonos imagery: indications from 2D transformations. *Photogrammetric Record*, 17(98):317-329.

Hanley, H., Yamakawa, T. and Fraser, C., 2002. Sensor orientation for high-resolution satellite imagery. *International Archives of Photogrammetry & Remote Sensing*, 34(1):69-75.

Haugerud, R.A. and Harding, D.J., 2001. Some algorithms for virtual deforestation (VDF) of LIDAR topographic survey data. *International Archive of Photogrammetry and Remote Sensing*, 34(3/W4):211-218.

Haverkamp, D. and Poulsen, R., 2003. Change detection using IKONOS imagery. *Proceedings of ASPRS 2003 Conference*, Anchorage, Alaska, 5-9 May 2003. CD-ROM.

Henricsson, O., 1998. The role of colour attributes and similarity grouping in 3-D building reconstruction. *Computer Vision and Image Understanding*, 72(2):163-184.

Holland, D., Guilford, B. and Murray, K., (eds) 2002. *OEEPE-Project on topographic mapping from high resolution space sensors*, no.44. Media-Print, Paderborn, Germany.

Holland, D. and Marshall, P., 2003. Using high-resolution satellite imagery in a well-mapped county. *Proceedings of Joint ISPRS Workshop on High Resolution Mapping from Space 2003*, Hannover, Germany, 6-8 October 2003. CD-ROM.

Hu, Y. and Tao, C.V., 2001. Updating solutions of the rational function model using additional control points and enhanced photogrammetric processing. *Proceedings of Joint ISPRS Workshop on High Resolution Mapping from Space 2001*, Hanover, 19-21 September 2001. CD-ROM.

Hu, X. and Tao, C.V., 2003. Automatic extraction of main-road centerlines from IKONOS and Quick-Bird imagery using perceptual grouping. *Proceedings of ASPRS 2003 Conference*, Anchorage, Anchorage, Alaska, 5-9 May 2003. CD-ROM.

Huising, E.J. and Pereira, L.M.G., 1998. Errors and accuracy estimates of laser data acquired by various laser scanning systems for topographic applications. *ISPRS Journal of Photogrammetry and Remote Sensing*, 53(5):245-261.

Jamet, O., Dissard, O. and Airault, S., 1995. Building extraction from stereo pairs of aerial images: accuracy and productivity constraint of a topographical production line. In A. Grün, O. Kübler, & P. Agouris (eds), *Automatic Extraction of Man-Made Objects from Aerial and Space Images*: 231-240. Birkhäuser-Verlag, Basel.

Jackson, A.M., 2003. Report for assessment of Burns straight line extraction algorithms using IKONOS imagery; ALFIE (Automatic Linear Feature Identification and Extraction) project, QinetiQ.

Jaynes, C., Stolle, T. and Collins. R., 1994. Task driven perceptual organization for extraction for rooftop polygons. *Proceedings of ARPA Image Understanding Workshop*.

Jensen, J.R., 1986. *Introductory Digital Image Processing*, Prentice-Hall, New Jersey, 379 pages.

Katzenbeisser, R.-I., 2003. About the calibration of LIDAR sensors. *Proceedings of ISPRS working group III/3 workshop on 3-D reconstruction from airborne laserscanner and InSAR data,* Dresden, Genrmany, 8-10 October 2003, 34(3/W13):59-64.

Kilian, J., Haala, N. and Englich, M., 1996. Capture and evaluation of airborne laser scanner data. *International Archives of Photogrammetry and Remote Sensing*, 31(B3):383-388.

Kim, J. R. and Muller, J-. P., 2002. 3D reconstruction from very high resolution satellite stereo and its application to object identification. *Proceedings of ISPRS technical Commission IV Symposium 2002 on Geospatial theory, Processing and Applications*, Ottawa, Canada, July 9-12, 2002, 34(4). CD-ROM.

Kim, T. and Muller, J.-P., 1995. Using image understanding fusion techniques. In A. Grün, E.P. Baltsavias & O. Henricsson (eds), *Automatic Extraction of Man-Made Objects from Aerial and Space Images (III)*: 221-230. Balkema Publishers, Lisse, the Netherlands.

Kim, T. and Muller, J.-P., 1999. Development of a graph-based approach for building detection. *Image and Vision Computing*, 17(1999):3-14.

Kim, Z., Huertas, A. and Nevatia, R., 2001. A model-based approach for multi-view complex building description. In A. Grün, E.P. Baltsavias & O. Henricsson (eds), *Automatic Extraction of Man-Made Objects from Aerial and Space Images (III)*: 181-194. Balkema Publishers, Lisse, the Netherlands.

Kraus, K. and Pfeifer, N., 1998. Determination of terrain models in wooded areas with airborne laser scanner data. ISPRS *ISPRS Journal of Photogrammetry and Remote Sensing*, 53:193-203.

Kraus, K. and Pfeifer, N., 2001. Advanced DTM generation from LIDAR data *International Archives of Photogrammetry and Remote Sensing*, 34(3/W4):23-30.

Lange, E., 1999. The degree of realism of gis-based virtual landscapes: Implications for spatial planning. In D. Fritsch & R. Spiller (eds), *Photogrammetric Week '99*:367-374. Herber Wichmann Verlag, Heidelberg.

Leberl, F., Walcher, W., Wilson, R. and Gruber, M. 1999. Models of urban areas for line-of-sight analyses. *International Archives of Photogrammetry and Remote Sensing*, 32(3/2W5):217-226.

Lee, D.S., Shan, J. and Bethel, J.S., 2003. Class-guided building extraction from Ikonos imagery. *Photogrammetric Engineering & Remote Sensing*, 69(2):143-150.

Lee, I. and Schenk, T., 2001. 3D perceptual organization of laser altimetry data. *International Archives of Photogrammetry and Remote Sensing*, 34(3/W4):57-65

Lee, I. and Schenk, T., 2002. Perceptual Organization of 3D Surface Points. *Proceedings of ISPRS Commission III, Symposium 2002 on Photogrammetric Computer Vision*, Graz, Austria, 9-13 September 2002, 34(3A):193-198.

Li, M., 1993. Minimum description length based 2D shape description. *Proceedings of 4th International Conference on Computer Vision*: 512-517, Berlin, Germany.

Li, S.Z., 2001. Markov random field modelling in image analysis. Springer-Veralg, Tokyo.

Lin, C., Huertas, A. and Nevatia, R., 1994. Detection of buildings using perceptual grouping and shadows. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*: 62-69.

Lin, C. and Nevatia, R., 1998. Building detection and description from a single intensity image, *Computer Vision Image Understanding*, 72:101-121

Lloyd, C.D. and Atkinson, P.M., 2002. Non-stationary approaches for terrain mapping and assessment of uncertainty in estimates. *Transactions in GIS*, 6:17-30

Lohmann, P., Koch, A. and Schaeffer, M., 2000. Approaches to the filtering of laser scanner data. *International Archives of Photogrammetry and Remote Sensing*, 33(B3):540-547.

Lumina, R., Shapiro, L., Zuniga, O., 1983. A new connected components algorithm for virtual memory computers. *Computer Vision, Graphics, and Image Processing*, 22:287-300.

Marshall, P., 2002. High resolution sensor data for topographic mapping. In D. Holland, B. Guilford, & K. Murray (eds), *OEEPE-Project on topographic mapping from high resolution space sensors*,. Media-Print, Paderborn, Germany. 44:31-52.

Masaharu, H. and Ohtsubo, K., 2002. A Filtering Method of Airborne Laser Scanner Data for Complex Terrain. *Proceedings of ISPRS Commission III, Symposium 2002 on Photogrammetric Computer Vision*, Graz, Austria, 9-13 September 2002, 34 (3B):165-169.

Mass, H.-G. and Vosselman, G., 1999. Two algorithms for extracting building models from raw laser altimetry data. *ISPRS Journal of Photogrammetry and Remote Sensing*, 54(1999):153-163.

Mass, H.-G., 2000. Least squares matching with airborne laserscanning data in a TIN structure. *International Archives of Photogrammetry and Remote Sensing*, 33(B3/1):548-555.

Mayer, H., 1999. Automatic object extraction from aerial imagery – a survey focusing on buildings. *Computer Vision and Image Understanding*, 74(2):138-149.

Mcintosh, K. and Krupnik, A., 2002. Integration of Laser-derived DSMs and matched image edges for generating an accurate surface model. *ISPRS Journal of Photogrammetry and Remote Sensing*, 23(3):242-256.

Mohan, R., & Nevatia, R. 1989. Using perceptual organization to extract 3-D structures. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(11):1121-1139.

Moons, T., Frere, D., Vandekerckhove, J. and Gool, L., 1998. Automatic modeling and 3D reconstruction of urban house roofs from high resolution aerial imagery. *Proceedings of the European Conference on Computer Vision.*

Morsdorf, F., Meier, E., Allgöwer, B. and Nüesch, D., 2003. Clustering in airborne laserscanning raw data for segmentation of single trees. *Proceedings of ISPRS*

working group III/3 workshop on 3-D reconstruction from airborne laserscanner and InSAR data, Dresden, Germany, 8-10 October 2003, 34(3/W13):27-33.

Muller, J.-P., Ourzik, C., Kim, T. and Dowman, I.J., 1997. Assessment of the effects of resolution on automated DEM and building extraction. In A. Gruen, E. P. Baltsavias & O. Henricsson (eds), *Automatic extraction of man-made objects from aerial and space images (II)*: 233-242. Basel, Birkhäuser.

Muller, J-.P., Kim, J-R. and Tong, L., 2001. Automated mapping of surface roughness and landuse from simulated and spaceborne 1m data. In A. Grün, E.P. Baltsavias & O. Henricsson (eds), *Automatic Extraction of Man-Made Objects from Aerial and Space Images (III)*: 369-380. Balkema Publishers, Lisse, the Netherlands.

Naylor, B.F., Amandatides, J., and Thibault, W., 1990. Merging BSP trees yields polyhedral set operations. *ACM Computer Graphics*, 24(4):115-124.

Noronha, S. and Nevatia, R., 1997. Building detection and description from multiple aerial images. *Proceedings of Conference on IEEE Computer Vision and Patter Recognition*: 588-594, San Juan, Puerto Rico.

Noronha, S. and Nevatia, R., 2001. Detection and modeling of buildings from multiple aerial images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(5):501-518.

OEEPE. 2000. Working Group on laser data acquisition. *ISPRS Congress 2000.* http://www.geomatics.kth.se/~fotogram/OEEPE/ISPRS_Amsterdam_OEEPE_presentaton.pdf.

Ordnance Survey, 2003. *OS MasterMap® Produce Guide.* http://www.ordnancesurvey.co.uk/oswebsite/products/osmastermap

Paparoditis, N., Maillet, G., Taillandier, F., Jibrini, H., Jung, F., Guigues, L. and Boldo, D., 2001. Multi-image 3D feature and DSM extraction for change detection and building reconstruction. In A. Grün, E.P. Baltsavias & O. Henricsson (eds), *Automatic Extraction of Man-Made Objects from Aerial and Space Images (III)*: 217-230. Balkema Publishers, Lisse, the Netherlands.

Papadaki, H., 2002. Report for establishment and quality control of stereo-model in SocetSet; Department of Geomatic Engineering: UCL, 2002.

Parish, B., 2002. Quality assessment of automated object identification in IKONOS Imagery. M.Sc. Thesis, University College London, London.

Park, W., Kwak, K., Su, H. and Kim, T.G., 2000. Line rolling algorithm for automated building extraction from 1-meter resolution satellite images. *Proceedings of the international symposium on remote sensing*: 31-36, Kyungju, 1-3 November 2000.

Pasko, M. and Gruber, M., 1996. Fusion of 2D GIS data and aerial images for 3D building reconstruction. *International Archives of Photogrammetry and Remote Sensing*, 31(B3):257-260.

Paterson, M.S. and Yao, F.F., 1989. Binary partitions with applications to hidden surface removal and solid modeling. *Proceedings of the 5th Annual ACM Symposium on Computational Geometry*: 23-32.

Pavlidis, T. and Liow, Y., 1990. Integrating Region Growing and Edge Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(3):225-233.

Petzold, B., Reiss, P. and Stossel, W., 1999. Laser scanning – surveying and mapping agencies are using a new technique for the derivation of digital terrain models. *ISPRS Journal of Photogrammetry and Remote Sensing*, 54:95-104.

Pfeifer, N. and Kraus, K., 1998. Interpolation and filtering of laser scanner data – implementation and first results. *International Archives of Photogrammetry and Remote Sensing*, Columbus, 32(3/1):153-159.

Pfeifer, N., Stadler, P. and Briese, C., 2001. Derivation of digital terrain models in the SCOP++ environment. Proceedings of OEEPE Workshop on Airborne Laserscanning and Interferometric SAR for Detailed Digital Terrain Models, Stockholm, Sweden.

Pyysalo, U. and Hyyppä, H., 2002. Reconstructing tree crowns from laser scanner data for feature extraction. *Proceedings of ISPRS Commission III, Symposium 2002 on Photogrammetric Computer Vision*, Graz, Austria, 9-13 September 2002, 34 (3B):218-226.

Qiu, G. and Sudirman, S., 2001. Color image coding, indexing and retrieval using binary space partitioning tree, Proceedings of IEEE ICIP 2001, 1:682 – 685

Radha, H.R., Leonardi, R., Vetterli, M. and Naylor, B.F., 1991. Binary Space Partitioning Tree Representation Of Images, Journal of Visual Communication and Image Representation, 2:201-221.

Rissanen, J., 1984. Universal coding, information, prediction, and estimation. *IEEE Transactions of Information Theory*, 30(4):629-636.

Roggero, M., 2001. Airborne Laser Scanning: clustering in raw data. *International Archives of Photogrammetry and Remote Sensing*, Annapolis, MD, 22-24 Oct, 2001, 34(3/W4):227-232.

Ryden, A. and Sjöed, J., 2002. High resolution sensor data for topographic mapping. In D. Holland, B. Guilford, & K. Murray (eds), *OEEPE-Project on topographic mapping from high resolution space sensors*, Media-Print, Paderborn, Germany, 44:31-52.

Schenk, T., 2001. Modeling and recovering systematic errors in airborne laser scanners. Proceedings of OEEPE Workshop on Airborne Laserscanning and Interferometric SAR for Detailed Digital Terrain Models, Stockholm, Sweden.

Schickler, W. and Thorpe, A., 2001. Surface estimation based on LIDAR. *Proceedings of ASPRS conference*, St. Louis Missouri, 23-27 April 2001. CD-ROM.

Schmid, C. and Zisserman, A., 1997. Automatic line matching across views. *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*: 666-671, 17-19 June, Puerto Rico.

Scholze, S., Moons, T. and Gool, V.L., 2001. A probabilistic approach to roof patch extraction and reconstruction. In A. Grün, E.P. Baltsavias & O. Henricsson (eds), *Automatic Extraction of Man-Made Objects from Aerial and Space Images (III)*: 195-204. Balkema Publishers, Lisse, the Netherlands.

Shufelt, J.A. and McKeown, D.M., 1993. Fusion of monocular cues to detect man-made structures in aerial imagery. *Computer Vision and Image Understanding*, I57(3):307-330

Shufelt, J.A., 1996. Exploiting photogrammetric methods for building extraction in aerial images. *International Archives of Photogrammetry and Remote Sensing*, 31(B6):74-79.

Shufelt, J.A., 1999. Performance evaluation and analysis of monocular building extraction from aerial imagery. *IEEE Pattern Analysis and Machine Intelligence*, 21(4):311-326.

Sithole, G., 2001. Filtering of laser altimetry data using a slope adaptive filter. International Archive of Photogrammetry and Remote Sensing, 34(3/W4):203-210.

Sithole, G., 2002. Filtering strategy: working towards reliability. *Proceedings of ISPRS Commission III, Symposium 2002 on Photogrammetric Computer Vision*, Graz, Austria, 9-13 September 2002, 34 (3A):330-335.

Sithole, G. and Vosselman, G., 2003a. Automatic structure detection in a point-cloud of an urban landscape. *Proceedings of 2nd Joint Workshop on Remote Sensing and Data Fusion over Urban Areas*, Berlin, Germany, 22-23 May 2003.

Sithole, G. and Vosselman, G., 2003b. Comparison of filtering algorithms. *Proceedings of ISPRS working group III/3 workshop on 3-D reconstruction from airborne laserscanner and InSAR data*, Dresden, Genrmany, 8-10 October 2003, 34(3/W13):71-78.

Sithole, G. and Vosselman, G., 2003c. Report: ISPRS Comparison of Filters. http://www.geo.tudelft.nl/frs/isprs/filtertest/Report05082003.pdf.

Smith, S.L., Holland, D.A. and Longley, P.A., 2003. Investigating the spatial structure of error in digital surface models derived from laser scanning data. *Proceedings of ISPRS working group III/3 workshop on 3-D reconstruction from airborne laserscanner and InSAR data*, Dresden, Genrmany, 8-10 October 2003, 34(3/W13):136-142.

Sohn, G., Dowman, I.J., 2001. Extraction of buildings from high resolution satellite data. In A. Grün, E.P. Baltsavias & O. Henricsson (eds), *Automatic Extraction of Man-Made Objects from Aerial and Space Images (III)*: 345-356. Balkema Publishers, Lisse, the Netherlands.

Sohn, G. and Dowman, I.J., 2002. Terrain surface reconstruction by the use of tetrahedron model with the MDL Criterion. *Proceedings of ISPRS Commission III,*

Symposium 2002 on Photogrammetric Computer Vision, Graz, Austria, 9-13 September 2002, 34 (3A):336-344.

Sohn, G. and Dowman, I.J., 2003. Building extraction using Lidar DEMS and Ikonos images. *Proceedings of ISPRS working group III/3 workshop on 3-D reconstruction from airborne laserscanner and InSAR data,* Dresden, Genrmany, 8-10 October 2003, 34(3/W13):167-173.

Space Imaging. 2003. IKONOS product guide. http://www.spaceimaging.com/whitepapers_pdfs/IKONOS_Product_Guide.pdf .

Steinle, E. and Voegtle, T., 2001. Automated extraction and reconstruction of buildings in laserscanning data for disaster management. In A. Grün, E.P. Baltsavias & O. Henricsson (eds), *Automatic Extraction of Man-Made Objects from Aerial and Space Images (III)*: 309-318. Balkema Publishers, Lisse, the Netherlands.

Suveg, I. and Vosselman, G., 2002. Automatic 3D reconstruction of buildings from aerial images. *SPIE Photonics West, Electronic Imaging*, 4661:59-69.

Tao, C.V. and Hu, Y., 2001a. A comprehensive study of the rational function model for photogrammetric processing, *Photogrammetric Engineering & Remote Sensing*, 67(12):1347-1357.

Tao, C.V. and Hu, Y., 2001b. The Rational function model – a tool for processing high-resolution imagery. *Earth Observation Magazine (EOM)*, 10(1):13-16.

TerraScan, 1999. TerraScan User's guide. http://terrasolid.fi/ftp/manuals/eng/tscan.pdf.

Thibault, W.C. and Naylor B.F., 1987. Set operations on polyhedra using binary space partition trees. *Computer Graphics*, 21(4):153-162.

Tokuta, A.O., 1994. Motion planning using binary space partitioning. *IEEE International Workshop on Intelligent Robots and Systems IROS'91*: 86-90, Osaka, Japan, November 1994.

Vosselman, G., 1999. Building reconstruction using planar faces in very high density height data. *International Archives of Photogrammetry and Remote Sensing*, 32:87-92.

Vosselman, G., 2000. Slope based filtering of laser altimetry data. *International Archives of Photogrammetry and Remote Sensing*, 33(B3), Amsterdam, 2000. CD-ROM.

Vosselman, G. and Maas, H.-G., 2001. Adjustment and filtering of raw laser altimetry data. Proceedings of OEEPE workshop on Airborne Laserscanning and Interferometric SAR for Detailed Digital Elevation Models. 40:62-72.

Vosselman, G. and Suveg, I., 2001. Map based building reconstruction from laser data and images. In A. Grün, E.P. Baltsavias & O. Henricsson (eds), *Automatic Extraction of Man-Made Objects from Aerial and Space Images (III)*: 231-242. Balkema Publishers, Lisse, the Netherlands.

Wack, R. and Wimmer, A., 2002. Digital terrain models from airborne laserscanner data – a grid based approach. *Proceedings of Photogrammetric Computer Vision ISPRS Commission III, Symposium 2002*, September 9-13, 2002, Graz, Austria:293-296.

Wack, R., Schardt, L., Barrucho, L., Lohr, U. and Oliveira, T., 2003. *Proceedings of ISPRS working group III/3 workshop on 3-D reconstruction from airborne laserscanner and InSAR data*, Dresden, Genrmany, 8-10 October 2003, 34(3/W13):40-46.

Wang, Z. and Schenk, T., 2000. Building extraction and reconstruction from lidar data. *International Archives of Photogrammetry and Remote Sensing*, 33(B3):958-964.

Schenk, T. and Csatho, B., 2002. Fusion of lidar data and aerial imagery for a more complete surface description. *International Archives of Photogrammetry and Remote Sensing*, 34(3A):310-317.

Wehr, U. and Lohr, U., 1999. Airborne laser scanning – an introduction and overview. *ISPRS Journal of Photogrammetry and Remote Sensing*, 54:68-82.

Weidner, U. and Förstner, W., 1995. Towards automatic building reconstruction from high resolution digital elevation models, *ISPRS Journal of Photogrammetry and Remote Sensing*, 50(4):38-49.

Wiman, H., 1995. Object related image segmentation using the minimum description length principle, *The 9th Scandinavian Conference on image Analysis*, pp 541-549

Wouters, W. and Bollweg, A., 1998. A detailed elevation model using airborne laser altimetry. *Geinformatics*, 9:6-9.

Yang, G.Z., Burger, P., Firmin, D.N. and Underwood, S.R., 1996. Structure adaptive anisotropic image filtering. *Image and Vision Computing,* 14(2):135-145.

Yoon, T., Kim, T., Park, W. and Kim, T.G., 1999. Building segmentation using an active contour model. *Proceedings of the ISPRS conference on sensors and mapping from space*, Hannover, 27-30 September 1999. CD-ROM.

Zhang, K., Chen, S., Whitman, D., Shyu, M., Yan, J. and Zhang, C., 2002. A progressive morphological filter for removing non-ground measurements from airborne LIDAR data. *IEEE Transactions on Geoscience and Remote Sensing*, 41(4):872-882.

Zhang, Q. and Wang, J., 2003. A rule-based urban land use inferring method for fine-resolution multispectral imagery. *The Canadian Journal of Remote Sensing*, 29(1):1-13.

Zhang, Z., Zhang, J. and Hu, X., 2001. Semi-automatic building extraction from stereo image pairs. In A. Grün, E.P. Baltsavias & O. Henricsson (eds), *Automatic Extraction of Man-Made Objects from Aerial and Space Images (III)*: 115-124. Balkema Publishers, Lisse, the Netherlands.

Zhu, S.C. and Yuille, A., 1996. Region competition: unifying snakes, region growing, and Bayes/MDL for multiband image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(9):884-900.

Ziebart, M. and Adhya, S., 2002. Report for Establishment of control and profile data; NextMap project; Department of Geomatic Engineering; UCL, 2002.